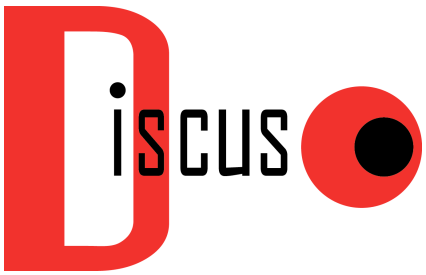


D 7.6

Core network optimization and resiliencey strategies

Dissemination Level:

- **Dissemination level:**

  PU = Public,
  RE = Restricted to a group specified by the consortium (including the Commission Services),
  PP = Restricted to other programme participants (including the Commission Services),
  CO = Confidential, only for members of the consortium (including the Commission Services)

Abstract:

This deliverable describes the optimization models and methods that are needed in order to: *(i)* optimize the design and operation of the DISCUS core network, and *(ii)* guarantee that the DISCUS architecture is able to guarantee to the provisioned services survivability in the presence of core network element failures.

The work presented in the document is divided into three main parts. In the first one the document starts by introducing a framework for evaluating the hardware cost (i.e., Chapter 2). This information is subsequently used as the basis for the design of single and multilayer core network solutions (i.e., Chapter 3 and 4). The second part of the deliverable (i.e., Chapter 5 and 6) focuses more on the network in operation, i.e., it addresses the problem of providing survivability against failures of core network elements in the presence of dynamic traffic. The last part of the deliverable addresses two additional aspects related to reliability. Chapter 7 provides an insight on what is the impact of energy saving mechanisms on the lifetime of core network devices, while Chapter 8 introduces a control architecture that can be implemented to support fast and accurate reaction to failures.

COPYRIGHT

The DISCUS Consortium consists of:

Participant Number	Participant organization name	Participant org. short name	Country
Coordinator			
1	Trinity College Dublin	TCD	Ireland
Other Beneficiaries			
2	Alcatel-Lucent Deutschland AG	ALUD	Germany
3	Nokia Siemens Networks GMBH & CO. KG	NSN	Germany
4	Telefonica Investigacion Y Desarrollo SA	TID	Spain
5	Telecom Italia S.p.A	TI	Italy
6	Aston University	ASTON	United Kingdom
7	Interuniversitair Micro-Electronica Centrum VZW	IMEC	Belgium
8	III V Lab GIE	III-V	France
9	University College Cork, National University of Ireland, Cork	Tyndall & UCC	Ireland
10	Polatis Ltd	POLATIS	United Kingdom
11	atesio GMBH	ATESIO	Germany
12	Kungliga Tekniska Hoegskolan	KTH	Sweden

Authors: (in alphabetic order)

Name	Affiliation
Norbert Ascheuer	ATESIO
Oscar Gonzalez de Dios	TID
Marija Furdek	KTH
Victor Lopez	TID
Deepak Mehta	UCC
Paolo Monti	KTH
Avishek Nag	TCD
Barry O'Sullivan	UCC
Cemalettin Ozturk	UCC
Luis Quesada	UCC
Christian Raack	ATESIO
Lena Wosinska	KTH

Internal reviewers:

Name	Affiliation
Andrea Di Giglio	TI
Marco Ruffini	TCD

Due date: 30.04.2015

Contents

Chapter 1

Introduction

Optical networks are exposed to a wide range of failures affecting single or multiple network components and disturbing a multitude of connections, possibly over greater geographical area. Failures can be caused, for example, by physical damage to the infrastructure - accidental optical fiber cuts due to construction work are the leading cause of network disruptions. Fatigue, mishandling, or environmental shocks can also result in faults of network components and devices. Aside from individual component failures, entire nodes can become disabled due to weather conditions, power outages, or natural disasters. Furthermore, the network infrastructure can also be targeted by malicious actions attempting to disrupt service or gain unauthorized access to information.

Due to the extremely high data rates and traffic volumes carried, nation-wide core networks must be able to provide quick and efficient recovery of affected connections. In order to guarantee service survivability in the presence of failures, network resilience must be taken into account both in the network design and during operation. In addition to be considered for deployment by the operators, the developed resiliency schemes must maintain high cost- and resource-efficiency.

The goal of this deliverable is to investigate means of increasing network resilience under a wide range of failures through judicious network design and connection provisioning in static and dynamic traffic scenarios. To be able to do so, it is necessary to first establish a framework for evaluating the hardware cost which then serves as the basis for network design. Network design further needs to consider the specific constraints of different network layers as well as interactions between the layers, leading to a very complex integrated planning problem. Furthermore, the control plane needs to support fast and accurate reaction to failures by triggering appropriate recovery mechanism.

Network resilience approaches in general rely on providing additional capacity in the network to be used as backup during failures. During network design, this redundancy can be provided by setting up the interconnections between nodes in a way which ensures the existence of physically disjoint routes through the network between all pairs of nodes. This means that additional physical links need to be set up in the network which are not vital to ensure connectivity under normal operating conditions, but are crucial to maintain connectivity in the event of failure. Due to the large sizes of the network, deciding which links to add while keeping the cost at a minimum and satisfying the signal reach entails a complex optimization procedure.

Redundant backup paths in the network which protect the primary paths of connections when they are affected by failures can be pre-planned, which is the underlying principle of protection approaches, or can be reserved dynamically, upon a failure, which is inherent for restoration. Protection strategies in the DISCUS topology can utilize the dually-homed access segment to obtain high connection availability under lower resource usage compared to the single-homing scenario. One of the goals of the work presented in the deliverable was to estimate the capacity overshoot needed to provide survivability in the dually-homed DISCUS reference topology when different design approaches are applied. In other words,

it is important to understand how much overprovisioning (in terms of extra WDM transponders) is needed to obtain a favourable tradeoff between resource usage and connection survivability.

Protection and restoration strategies can also be combined into a hybrid approach to achieve a resource-efficient increase of network resiliency under overlapping failures of more than one link. Under dynamic scenarios, resilience of optical cloud services can significantly benefit from the concept of service relocation, where the service is moved from one data center to another in the occurrence of a failure to allow for greater flexibility in backup path provisioning.

Survivability from failures of node components can take place at the node level as well without triggering network level recovery, provided that redundant components are placed inside the node and that the node architecture supports flexible operation. Reconfigurable Add Drop Multiplexers (ROADMs) implemented by Architecture on Demand (AoD) represent a promising approach for this functionality which could alleviate the resource consumption burden of failure recovery at the network level.

Survivability approaches which protect from component failures might not provide protection from deliberate, attack-like events in the network because both the primary and the backup path might be affected by the attack. Thus, such approaches need to identify the potential risk of the primary and backup path of connections being simultaneously affected by an attack and establish the paths in a way which reduces that risk, while maintaining resource-efficiency of conventional, failure-protection approaches.

Finally there is additional survivability aspect that is important to highlight, i.e., the impact of energy saving mechanisms on the lifetime a device. In fact a possible drawback of a green approach is that frequent on/sleep switching may negatively impact the failure rate performance of a device, and consequently increase its reparation costs. In particular, it is important to make sure that the potential savings brought by a reduced power consumption level are not lower than possible extra reparation costs caused by a reduced lifetime.

While some of the survivability approaches presented in this document focus on the DISCUS reference topology, others have been tested on a variety of reference topologies from the literature in order to gain a comprehensive insight into their behaviour. This thorough assessment of their performance will allow us to select the most promising approaches for the consolidated network design. This deliverable is organized as follows.

Chapter 2 provides a consolidated model of cost and hardware including core photonic switching, signal regeneration and Raman amplification. A dimensioning exercise is also provided to provide insight into optimization model parametrization.

The detailed model presented in Chapter 2 then serves as the basis for multi-layer network design study given in Chapter 3. Instead of adopting a common bottom-up approach of decoupling different network layers, which may lead to sub-optimal results, the work in Chapter 3 follows an integrated approach which allows for greater flexibility during optimization of the virtual topology and installation cost.

Chapter 4 focuses on resilient network planning by first incorporating resilience and signal reach constraints into network design. The work further investigates methods of providing node-level recovery of component faults in synthetic programmable ROADMs implemented by AoD, proposes an approach for dedicated path protection which minimizes the number of connections unprotected from physical-layer attacks aimed at service disruption, and studies the benefits of utilizing dual homing in protection from core link failures.

Chapter 5 investigates a hybrid approach for resilient service provisioning which combines path protection with path restoration. It then analyzes the benefits of placing redundant modules inside AoD-based ROADMs in achieving a beneficial tradeoff between connection availability and cost. Further, it develops an approach for restoring optical cloud services based on service relocation.

Chapter 6 deals with the problem of finding a favourable tradeoff between network overdimensioning and the resulting network survivability performance.

Chapter 7 investigates how energy saving mechanisms based on frequent on/sleep switching may impact the failure rate (i.e., lifetime) performance of a device.

Chapter 8 presents the interaction of the control plane in resilience mechanims in centralized and distributed scenarios.

Chapter 2

Cost modeling

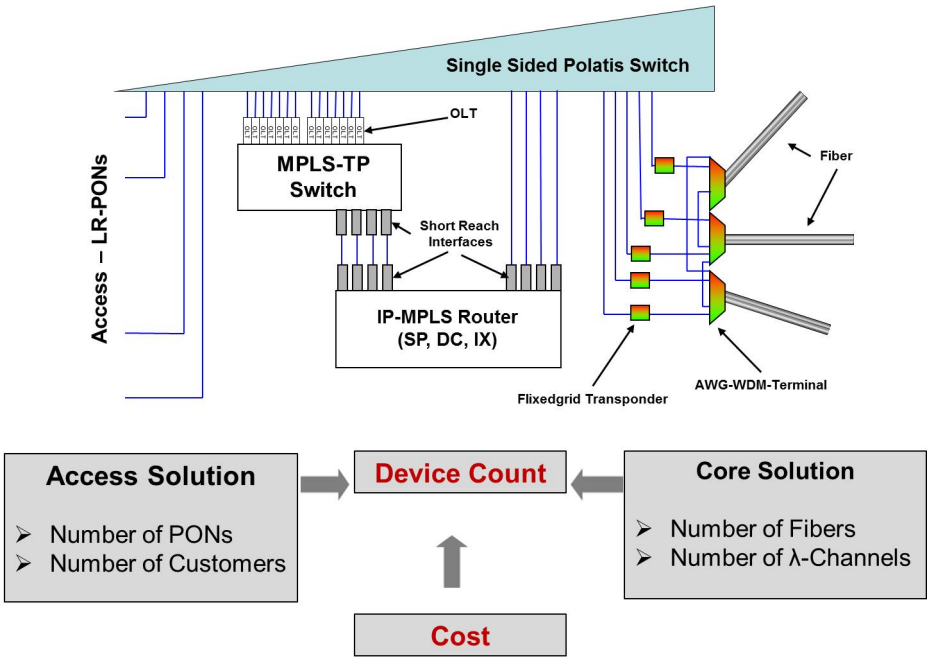


Figure 2.1: Cost evaluation based on network dimensioning. Simplified MC-node model: MPLS-TP, IP, and Photonic switching. IP functionality is needed only for service nodes such as for data centers (DC) and internet exchanges (IX). For optimization we simplify and assume a single MPLS-TP, a single IP, and a single Polatis switch. This is feasible as most of the cost comes from the port count.

In this chapter, we will introduce a cost and hardware model that we will use for any optimization related to the core network. This cost model is an extension of the tentative model presented in Deliverable D2.6 [6]. Starting from the model in [6], we mainly consolidated the cost for core photonic switching and added a model for signal regeneration and Raman amplification. Our model is in large parts based on the capex-models developed in the IDEALIST project, see [3]. We adapted the model following the DISCUS architecture. In particular we add cost values for the Polatis switch (see [8]) as well as for Raman amplifiers and the flex-grid signal types used within DISCUS (see [4, 10]). The flex-grid transponder model has been consolidated compared to Deliverable D2.6 [6].

Recall that this hardware model is designed to be used for optimization, that is, to parametrize objective functions of optimization models and to evaluate the cost of optimization solutions. In this respect, it only reflects a coarse view on the capital expenditures with a clear focus on the main cost-drivers. It cannot replace a detailed cost and cash-flow analysis. Such an analysis will be reported in Deliverable D2.8 [11].

As already mentioned in [6], for optimization it is feasible to ignore any cost that is independent of the solution, e.g. cost that is proportional to the number of customers, a constant.

In many cases, it is even feasible to further simplify and combine network elements to blocks of equipment introducing the notion of link or node *designs*. This will be done for instance with the core photonic switching elements below similar to what has been done in [82] and [3].

Tables 2.1, 2.2, and 2.3 present our cost and hardware model for electronic switching and photonic switching at the MC node, and for the Core fiber link, respectively. All cost values are given both in EUR and in the IDEALIST cost unit (ICU), where

$$1 \text{ ICU} = €\,50{,}000. \tag{2.1}$$

defines the conversion factor used.

All interfaces in this chapter are bidirectional. Even if we speak of a fiber line interface in the following, or simply a fiber we refer to a fiber pair in reality (one fiber for each direction).

2.1 Electronic switching at the MC node

The MC node hardware model in Table 2.1 provides cost values for the electronic switching elements at the MC node, see [7] and Figure 2.2.

As mentioned above, we partially made use of the models developed in the IDEALIST project and published in [3]. We reuse the 2015 models for IP-MPLS (router, cards) and MPLS-TP (switches, cards) with a restriction to 400G slot capacities and 40G, 100G, and 400G ports.

Type	Provides		Cost in €	Cost in ICU
IP-MPLS router	16	400G Slots	215,000	4.3000
	32	400G Slots	1,143,500	22.8700
	...	...	...	...
	1152	400G Slots	416,451,000	8329.0200
	...	...	...	...
IP-MPLS card	10	40 GE ports	128,000	2.5600
	4	100 GE ports	144,000	2.8800
	1	400 GE port	137,000	2.7400
MPLS-TP switch, 400G slots	16	400G Slots	192,000	3.8400
	32	400G Slots	384,000	7.6800
	...	...	...	...
	112	400G Slots	1,344,000	26.8800
	...	...	...	...
MPLS-TP line-card	10	40G ports	43,360	0.8672
	4	100G ports	54,200	1.0840
	1	400G port	60,680	1.2136
Transceiver, Grey, Short Reach	1	40G port	400	0.0080
	1	100G port	1,600	0.0320
	1	400G port	4,000	0.0800

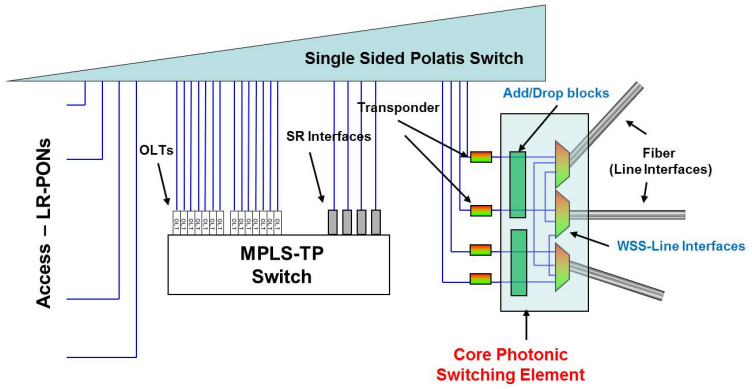
Table 2.1: MC-node hardware and cost model based on the values defined in [82] and [3]. We added OLT cards for MPLS-TP switches with a cost based on the cash-flow modeling in WP2 (8 port card has been scaled to 40 port card).

The node model assumes the L2, L3 routing and switching elements to be organized in three levels: chassis, cards, and transceivers, see [6, Figure 20]. The chassis is characterized by its capacity in terms of slots, the cards in terms of throughput and type and number of ports, and the interfaces in terms of their client rate. Line-Cards are supposed to require one slot in the chassis. We assume 400G slots for IP-MPLS as well as MPLS-TP. All cards may be equipped with short reach transceivers which in turn can be connected to transponders. We do not assume colored long reach transceivers here for simplicity. In fact, we may assume a colored interface at the cost of a short reach interface plus a transponder. For the cost of transponders see the optical equipment below.

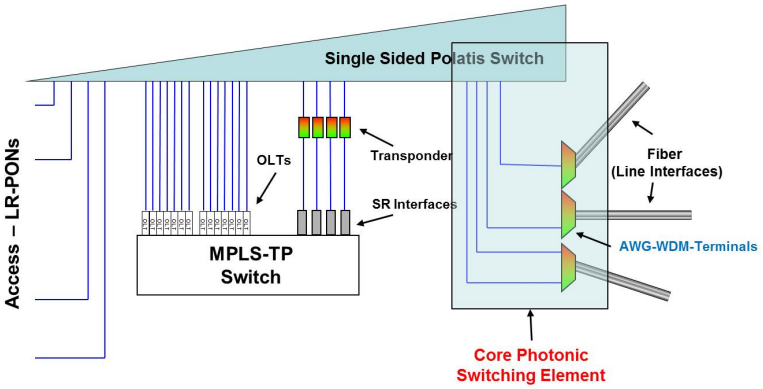
Type		Provides	Reach	Cost in €	Cost in ICU
Polatis Switch	100	Fiber ports		17,063	0.3413
	...	...		...	...
	400	Fiber ports		34,650	0.6930
	...	...		...	...
Fixed Grid					
Transponder	1	40G port	2500 km	24,000	0.4800
	1	100G port	2000 km	50,000	1.0000
	1	400G port	150 km	68,000	1.3600
WDM Terminal (AWG+Interleaver+OLA)	1	Fiber port		48,000	0.9600
Flexible Grid					
Transponder	1	40G port	2430 km	47,500	0.9500
	1	2x100G port	2430 km	150,000	3.0000
	1	100G port	1170 km	50,000	1.0000
	1	2x100G port	500 km	55,000	1.1000
	1	400G port	1170 km	150,000	3.0000
	1	2x400G port	500 km	160,000	3.2000
WDM Terminal (WSS + OLA)	1	Fiber port		78,000	1.56
Add Drop Block	16	Add Drop ports		32,000	0.64

Table 2.2: Hardware and cost model for optical core equipment. We assume regenerators at the cost of 1.6 times the corresponding transponder. Regenerators double the reach of the respective signal. The cost values for WDM terminals combine cost numbers from the IDEALIST model [3], which states 1 OLA = 0.3 ICU, 1 WSS (1x20/20x1) = 0.48 ICU, 1 AWG = 0.07 ICU, and 1 Interleaver = 0.04 ICU.

We will assume MPLS-TP switches at all MC nodes. For simplicity, in optimization we will not distinguish core-side and access-side switches. IP-MPLS routers exist only at service nodes, that is, IP-nodes of service providers (SP), data-center (DC), or internet peering points (internet exchanges, IX) see [6, Section 2.3] and Figure 2.1 compared to Figure 2.2(b). We remark that IP-routers with a slot count of more than 16 are multi-chassis routers. There is a significant cost increase from the single chassis router (16 slots) to the 2-chassis router with 32 slots. However the cost for larger routers then increases roughly linearly and can be computed with a simple formula based on the number of required slots, see [3] and [82].



(a) Switching using Optical Cross Connected based on WSS line interfaces



(b) Switching using AWG based Mux/Demux and the Polatis switch

Figure 2.2: Two options for core photonic switching

2.2 Photonic switching at the MC node

Table 2.2 aims at providing the optical switching hardware and cost. As explained in detail in [8] and [12] there are two options for core photonic switching at the MC node compatible with the DISCUS architecture.

WSS line interfaces

Option (a) as indicated in Figure 2.2(a) and 2.3 is available for both flex-grid and fixed-grid WDM. It is based on optical cross connects built from WSS (wavelength selected switches) and Add/Drop blocks with 16 Add/Drop ports each. In this case, the Polatis switch is not needed for switching core-to-core lambdas. It is used only to have more flexibility in switching wavelength services between the access network and the core network. We will refer to the fiber terminating elements as *WDM terminals*. These include the switches and amplification. For Option (a) it holds:

$$1 \text{ WDM Terminal} = 2 \text{ WSS} + 2 \text{ OLA}. \tag{2.2}$$

We have to provide 1 WDM Terminal for every fiber line interface at the MC node. In addition we need an appropriate number of Add/Drop ports, one for each terminated lambda

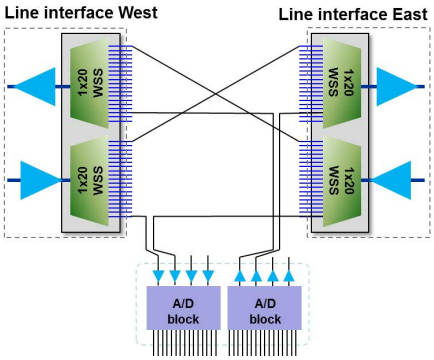


Figure 2.3: 2-degree core photonic switch based on (flex-grid) WSS line interfaces

channel. If f is the number of fiber ports and n the number of terminating channels, then the cost of the core photonic switching element for Option (a) amounts to

$$1.56 \cdot f + 0.64 \cdot \lceil \frac{n}{16} \rceil \text{ ICU}, \tag{2.3}$$

where 1.56 is the ICU cost for on WDM terminal based on WSS and 0.64 is the cost in ICU for one Add/Drop block with 16 ports. If n is the number of terminating channels, then we need

$$n_{polatis} = 2 \cdot n \tag{2.4}$$

ports at the Polatis switch.

AWG Mux/Demux and Polatis switch

Option (b) for photonic switching as indicated in Figure 2.2(b) is based on AWG Mux/Demux elements to (de)multiplex the WDM signals and the Polatis switch to cross-connect the individual lambda signals. However, this option with relatively moderate switching cost is available only for fixed-grid WDM systems. Again we refer to the fiber terminating element including amplification as *WDM terminal*. In case of Option (b) we have:

$$1 \text{ WDM Terminal} = 4 \text{ AWG} + 2 \text{ OLA} + 2 \text{ Interleaver}. \tag{2.5}$$

We have to provide 1 WDM Terminal for every fiber line interface at the MC node and no additional Add/Drop blocks. Instead an appropriate number of ports at the Polatis switch have to be provided. If f is the number of fiber ports and n the number of terminating channels, then we need

$$n_{polatis} = 176 \cdot f + 2 \cdot n \tag{2.6}$$

ports at the Polatis switch. The single-sided Polatis switch has cost

$$0.0224 + n_{polatis} \cdot 0.00118 \text{ ICU}, \tag{2.7}$$

where $n_{polatis}$ is the number of fiber ports at the Polatis switch. We will assume the Polatis switch to be used for access side and core side switching.

Type	Provides	Cost in €	Cost in ICU	Unit
Duct	Duct-Space	66,300	1.3260	Km
Cable	276 Fibers	7,859	0.1572	Km
Cable	240 Fibers	7,145	0.1429	Km
Cable	192 Fibers	6,145	0.1229	Km
Cable	144 Fibers	5,145	0.1029	Km
Cable	96 Fibers	4,145	0.0829	Km
Cable	48 Fibers	3,145	0.0629	Km
Cable	24 Fibers	2,716	0.0543	Km
Cable	12 Fibers	2,430	0.0486	Km
Raman amplifier	80 km Raman amplifier	30,000	0.6000	Piece
Optical Line Amplifier (OLA)	80 km EDFA amplifier	15,000	0.3000	Piece
Digital Gain Equalizer (DGE)	320 km equalizer	8,000	0.1600	Piece

Table 2.3: The cost per km of each individual fiber is between €28 (276-fiber cable) and €43.0 (96-fiber cable). Ignoring smaller cables it may hence be feasible to assume an average fiber cost of €40.0 per km.

Transponders and regeneration

Transponders (Fixedgrid and Flexgrid) are used on both sides of every core optical channel in combination with a short-reach transceiver interface at the MPLS-TP (or IP) switches. Depending on the signal, transponders have a certain signal-reach, see Table 2.2. The reach can be extended by regenerating the signal at the MC node (after add/drop) which is done using regenerators. Each transponder has a corresponding regenerator, which allows to regenerate the signal without going to the electrical switches. That is, the use of a regenerator doubles the reach of the respective signal types. We will assume regenerators to have a cost of 1.6 times the cost of the corresponding transponder.

Some of the transponders in Table 2.2 provide 2 client interfaces doubling the bitrate capacity. Clearly, these need two transceivers on each side.

2.3 Fiber link cost

For most of the studies in this Deliverable we assume spare dark fiber to be available to the operators at no installation cost. However, even in the spare fiber scenario using a core fiber incurs cost. First of all Optical Line Amplifiers (OLA) have to be provided every 80 kilometers and secondly, Digital Gain Equalizers (DGE) are needed every 320 kilometers of the fiber span. OLAs and DGE are already provided at all network nodes (included in the cost for WDM Terminals, see above). That is, a fiber with length k km between two MC nodes incurs a cost of

$$0.3 \cdot \lfloor \frac{k}{80} \rfloor + 0.16 \cdot \lfloor \frac{k}{320} \rfloor \text{ ICU}, \tag{2.8}$$

according to the cost for OLA and DGE in Table 2.3. Moreover, notice that the cost for line termination can be mapped to the fiber cost in all optimization models. That is, for each individual fiber, the cost for two WDM terminals (one on each end) can be added to the cost for OLA and DGE.

If instead a green-field scenario is considered we add cable installation and duct build cost. For cable installation we use the values provided by Table 2.3, which are identical to those used for the backhaul and E-side of the DISCUS network, see [6]. In addition we assume a duct build probability of 5%, that is, for each used fiber link in the network (independent of the number of fibers in use) we assume 5% of €66,300 = €3,315 per km duct build cost.

2.4 A dimensioning exercise

In the following we will use the introduced hardware and cost model to dimension the equipment of a single MC node. This helps us understand how to parametrize the objective functions of our optimization models.

Independent of the actual hardware and cost-model, the outcome of an optimization of the access network is the number of LR-PONs connected to every MC node. In fact, for core network optimization, we start from a given set of MC node locations with already connected customers and LR-PONs. Similarly, core network optimization returns the number of terminating fibers and the number of terminating channels at the MC node (independent of hardware and cost).

It turns out that the main cost incurring at an arbitrary MC node can be estimated starting from these three figures: (i) connected LR-PONs, (ii) terminating fibers, and (iii) terminating channels, see Figure 2.1.

Let us first assume the considered MC node does not provide data center or peering point services as the one in Figure 2.1. Let us further assume that optimization returns the following figures for one of the MC nodes:

- p connected LR-PONs
- f connected fibers
- n_{400} 400G channels, n_{100} 100G channels, n_{100} 40G channels.

In this case the cost of the MC node is estimated by counting the necessary network elements as follows. Electronic equipment depends on the number and type of terminated channels (core side) and the number connected LR-PONs. Transceivers, line-cards, and

Element	Number
40G transceiver	n_{40}
100G transceiver	n_{100}
400G transceiver	n_{400}
40G MPLS-TP line-card	$c_{40} = \lceil \frac{n_{40}}{10} \rceil$
100G MPLS-TP line-card	$c_{100} = \lceil \frac{n_{100}}{4} \rceil$
400G MPLS-TP line-card	$c_{400} = n_{400}$
OLT line-card	$c_{40} = \lceil \frac{p}{40} \rceil$
400G MPLS-TP slots	$c_{40} + c_{100} + c_{400} + c_{40}$

Table 2.4: Dimensioning: Electronic equipment at the MC node for fixed-grid WDM

the MPLS-TP switch are dimensioned accordingly. The resulting dimensioning of the MC node electronics can be found in Table 2.4.

For the optical equipment we have to count the transponders necessary to terminate the channels and the WDM terminals to terminate the fibers. In addition the Polatis switch has to be dimensioned based on the number of fibers and channels. See Table 2.5 and Table 2.6 for the resulting dimensioning of the MC node optics for fixedgrid and flex-grid WDM systems, respectively. Recall that in case of flex-grid WDM systems the count of the Polatis switch ports changes as well as the cost for the WDM terminal. In addition, we have to provide an appropriate number of Add/Drop blocks, see Table 2.6.

Element	Number
WDM Terminal (AWG)	f
40G transponder fixed grid	n_{40}
100G transponder fixed grid	n_{100}
400G transponder fixed grid	n_{400}
Polatis switch fiber ports	$p + 176f + 2(n_{40} + n_{100} + n_{400})$

Table 2.5: Dimensioning: Optical equipment at the MC node for fixed-grid WDM

Element	Number
WDM Terminal (WSS)	f
Add/Drop blocks	$\lceil \frac{(n_{40}+n_{100}+n_{400})}{16} \rceil$
40G transponder flex-grid	n_{40}
100G transponder flex-grid	n_{100}
400G transponder flex-grid	n_{400}
Polatis switch fiber ports	$p + 2(n_{40} + n_{100} + n_{400})$

Table 2.6: Dimensioning: Optical equipment at the MC node for flex-grid WDM

Based on these device counts and the cost values in Table 2.1 and 2.2 we can easily provide a reasonable estimate for the cost of a single MC node. Notice that for the cost of the transponders we need to know the actual signal type in addition to the count of the interfaces. To determine the total cost of the core network we aggregate the cost at all MC nodes and add the cost for fibers following the remarks above and values provided in Table 2.3.

Chapter 3

Multi-layer network design

In this chapter, we will study the cost and scalability of core networks based on the optical island concept. In particular, we will compare optical islands with hierarchical architecture concepts that are based on grooming traffic before entering a smaller inner core network. We will prove that optical islands are future-proof in the sense that they are the most cost-effective with respect to increasing traffic volumes. For selected network scenarios, we will also present the traffic volume threshold above which optical islands are less expensive than grooming architectures. All cost values are based on the cost model presented in Chapter 2.

Before we show our computational results and findings we have to introduce the required mathematical concepts and algorithmic machinery.

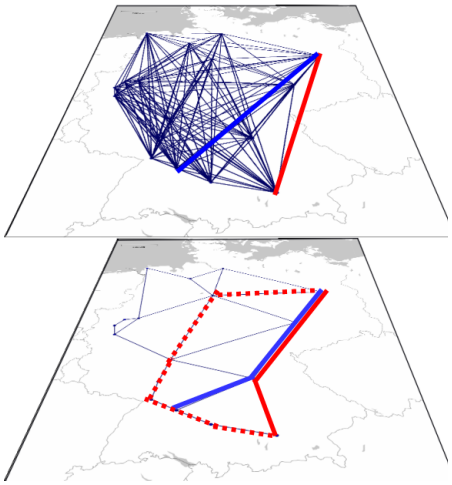


Figure 3.1: Client-Server relation in multi-layer networks: Links of the (virtual) client layer are realized by paths in the (physical) server layer. Each of the paths consumes capacity in the server.

In practice, telecommunication core networks consist of a stack of technologically different network layers, which are embedded into each other following a client-server relation. In the following we will think of client layers being 'on top' server layers. The links of a client layer can be seen as requests or demands. These request are realized by paths in the server layer. The server layer has to provide the necessary capacities. The realized capacities form links in the server layer, which in turn become requests for the next underlying network layer in the stack. In this respect each layer may be server and client at the same time.

Each layer is defined by its provided capacity unit (to realize requests from the above client) and its consumed capacity unit (the request to the next layer below). Typical capacity units are Gbps, channel, fiber, cable, ducts. Capacities are typically provided as multiples of a base unit, e.g. multiples of 40, 100, 400 Gbps, or multiples of 88-slot DWDM systems (fibers). Each layer may also restrict the possible realization of its demands, e.g. it may

Layer	Provides capacity in terms of	Requests capacity in terms of
Service		Traffic in Gbps
Virtual	Gbps (40G, 100G, 400G)	Channel slots (1,2,4)
Physical	Channel slots (88, 120)	Fibers
Cable	Fibers	Ducts
Duct	Ducts	Trenches
Trench	Trenches	

Table 3.1: The network layer structure with provided and consumed resources. For core network design we will ignore Trench and Duct layer and assume the Cable layer with spare fibers to be given.

force a single path realization instead of a splitting of the request across multiple paths. It may also claim a certain level of survivability of the realization of its requests.

This strong coupling of capacities embedded in capacities across multiple layers yields one of the most challenging dimensioning problem in combinatorial optimization often called *multi-layer network design*: The task is to dimension all network layers in such a way that all requests can be realized across all layers, while minimizing the cost for all resources. We refer to [72] for a detailed mathematical analysis of the structure of the two-layer version of this problem, also see [77].

3.1 DISCUS layers

From a schematic and mathematical point of view we may distinguish the following network layers within the DISCUS core network:

Service layer

Based on the traffic modeling in Deliverable D2.4 [5], Deliverable D2.6 [6] and D2.8 [11] (forthcoming) we may assume that we are given a demand matrix that defines for each pair of MC nodes a traffic demand in Gbps. At this point we aggregate the traffic for different service characteristics (internet exchange traffic, data center traffic, peer-to-peer traffic) to a single *service link*. A service link is realized by paths in the underlying electronic switching layer and consumes Gbps.

Virtual layer

For simplicity, we will refer to the electronic switching layer (IP/MPLS-TP) as the *virtual layer*. The virtual layer provides capacity in terms of 40Gbps 100Gbps, or 400Gbps links, see the hardware and cost model 2. Each *virtual link* (a 40G,100G, or 400G link) depending on the signal type (the modulation format) requests a certain channel slot capacity in the underlying physical fiber layer. The individual slot demands of each signal type are stated in [10, Table 1].

Physical layer

For simplicity, we will refer to the optical transport, WDM or fiber layer as the *physical layer*. The physical layer provides channel slot capacity in terms of fibers. In fixed grid scenarios we assume fibers with 88 slots. In flex-grid scenarios a fiber has 120 slots (37.5 GHz spacing), see [4]. Fibers consume cable capacity.

Cable, duct, and trenching layers

The cable deployment layer provides fiber capacity. We distinguish cables of different sizes, see Table 2.3. Clearly, cables need duct space. The duct layer provides cable capacity and might need trenching.

The different layers and their resources are summarized in Table 3.1. Throughout the rest of this section we will ignore trench capacities and duct capacities. We will further assume the core cable layer with spare fibers to be given as input, see below. That is, we will concentrate on the design of the virtual and the physical layer based on a given traffic matrix (the service layer), which is a two-layer network design problem.

3.2 The DISCUS architecture: Optical islands

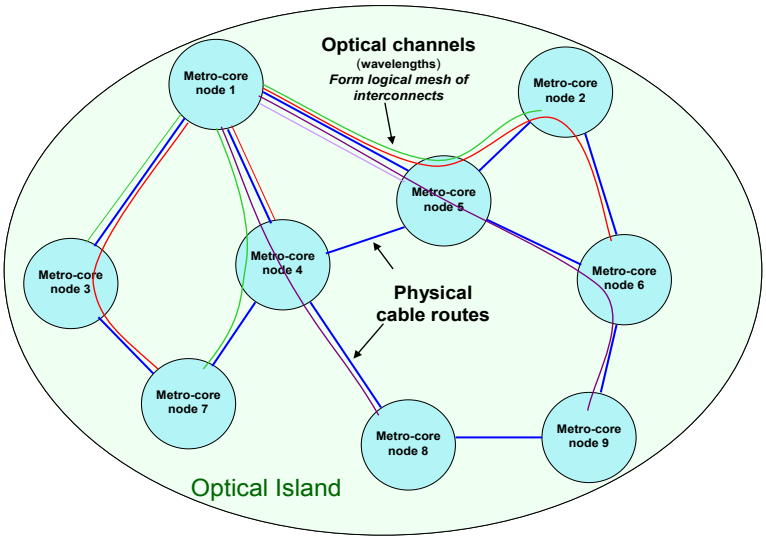


Figure 3.2: Optical islands: The cable links (with spare fibers) are in blue. The colored paths are optical channels, that is, virtual links and their realizations in the fiber layer. In the optical island concept we assume that each service link is realized by direct virtual links (single-hop virtual paths). It follows that the virtual layer becomes a full-mesh of optical channels. In this figure we see only a subset of the necessary virtual links.

In most practical situations there is a serious of constraints to be taken into account with respect to the layer structure. In particular there are restrictions on the allowed path realizations of a given client link. The DISCUS architecture introduces the following constraints on the client link realizations:

- *Service link realizations:* In a pure optical island, each service link is realized as a single-hop virtual path. That is, the service traffic between two MC nodes is sent directly using a direct optical channel connection (or multiple such channels). With this definition of an optical island there is an optical channel for each service link. Since we expect a service link (a demand) for each pair of MC nodes, the virtual layer becomes a full-mesh network. This results in a strong restriction on the possible virtual topologies and network architectures. Below we will study under which conditions this solution is cost-effective. In particular, we will compare optical islands with architectures that allow for grooming at intermediate MCs and their MPLS-TP or IP switches. Grooming may take place at all inner nodes of a virtual path.
- *Virtual link realizations:* Optical channels (virtual links) are realized as paths in the physical fiber layer. For these realizations there is a reach limit coming from the signal types and the corresponding transponders. Within DISCUS we assume 3 signal-types with reaches of 500km (400G), 2000km (100G), and 2500km (40G) for fixedgrid systems and 6 signal-types with reaches between 500 and 2430km for flex-grid systems. To realize an optical channel with a certain capacity and reach we have to provide the corresponding transponder on both sides of the virtual link, see Table 2.2. The use of regenerators doubles the reach of an optical channel but adds the additional cost of one regenerator to the optical channel.
- *Physical link realizations:* in general, any path in the given cable layer can be seen as a possible fiber link. For simplicity and flexibility, we will assume that both the cable links itself and the realization of fiber links do not contain intermediate MC nodes. That is, if a cable or fiber passes an MC node it is also terminated at that node. To terminate a fiber we need a WDM terminal at the MC node. Clearly, it might decrease the cost if a fiber is not terminated at an MC node. However, it removes flexibility since optical channels cannot be switched without terminating the fiber.

3.3 Data: MC node distributions, cable networks, traffic matrices

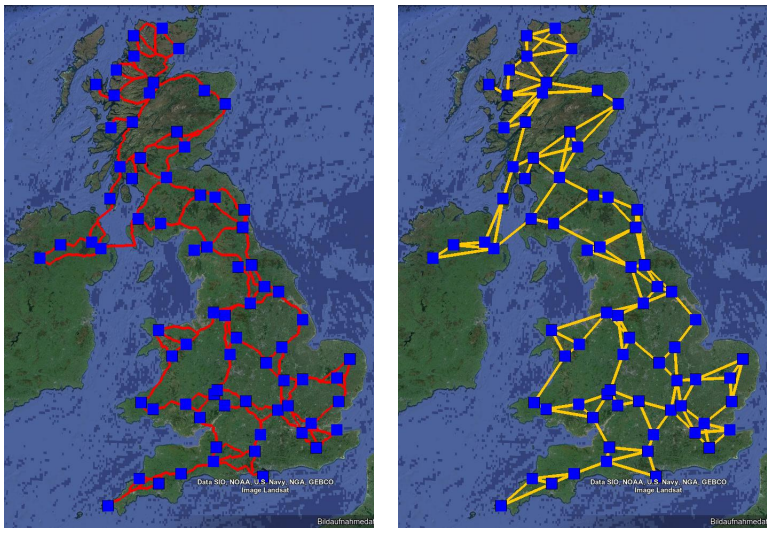
MC node distributions

The end-to-end optimization process as introduced in Deliverable D2.6 [6] decomposes access network and core network optimization. For core network optimization we assume a given set of MC nodes M. To test our multi-layer network dimensioning tool we use a series of different MC node distributions based on different reference networks (different countries) and different additional assumptions, e.g. assumptions on the maximum reach of the LR-PON.

Table 3.2 summarizes the instance we use for computational studies within this chapter. In all cases the given MC node distributions allow for a dual homing of all LE sites by (maximal) disjoint fiber connections within the reference fiber network, see [6]. The number of MC nodes can be considered to be the minimum under the given constraints. For an introduction to the reference networks for the UK, Italy, and Spain we also refer to [6].

Instance Country	$\max_{Km}$	# MC nodes	# Fiber links
UK	110 km	75	137
Italy	115 km	116	219
Spain	115 km	179	321

Table 3.2: Instances used in this chapter with the number of MC nodes and the number of potential fiber links. Instances are characterized by their reference network (country) and the maximum LE to MC distance $\max_{Km}$ used to calculate the MC node distribution, see [6].



(a) (b)

Figure 3.3: UK 75 MC node instance: (a) Cable reference (b) Potential fiber links

Cable networks

For the studies in this chapter we assume a *brown-field* scenario. That is, the core network is deployed on top of an existing cable and fiber-network. In this respect, we assume that cables with sufficient spare fibers exist. However, we have no access to the cable networks of operators in the UK, Italy, or Spain to build cable references along MC node distributions from Table 3.2.

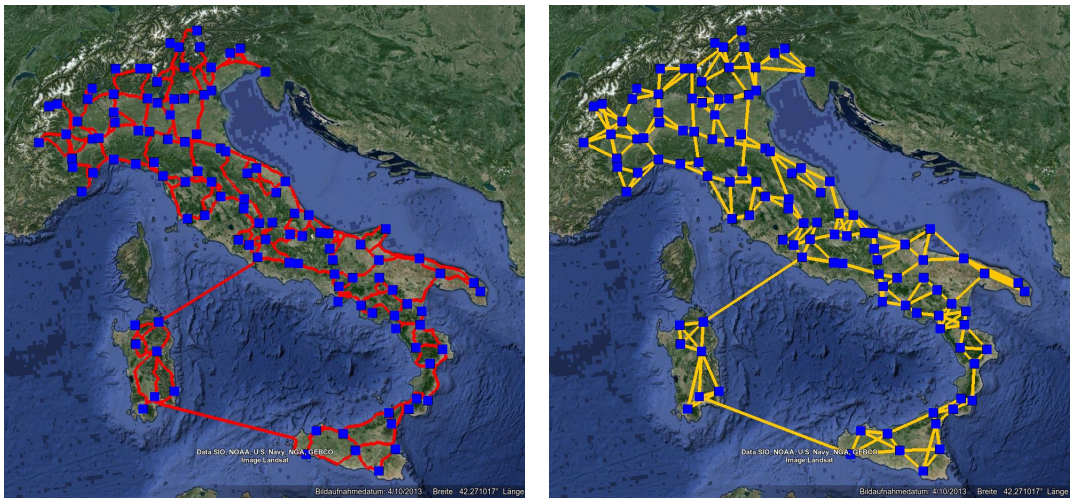
To come up with a cable and fiber reference network we roughly follow the first step of the solution approach presented Deliverable D7.4 [10, Chapter 4.1], the “Core Fiber Network Design”, also see Chapter 4.1. That is, we interconnect neighbored MC nodes with cable routes that follow short paths in the reference networks. Our aim is a reasonably connected cable and fiber network for each of the scenarios in Table 3.2.

In contrast to [10, Chapter 4.1] and the approach taken in Chapter 4.1 we use a fast heuristic based on the following steps:

1. We start from a reference network $N = (V, F)$, a given MC node distribution: $M \subset V$, and an empty set of fiber links E in the fiber network $G = (M, E)$.

2. Connect each MC to its 3 closest neighbor MCs via a shortest path in the reference network. Add the corresponding fiber links to E.

3. Take the pair of MC nodes (m_1, m_2) with the largest shortest path distance $l_G(m_1, m_2)$ in G. If G is not yet connected, then $l_G(m_1, m_2)$ is infinite.
4. Let $l_N(m_1, m_2)$ be the shortest path distance in the original reference network N.
5. If $l_G(m_1, m_2) \leq 2 \cdot l_N(m_1, m_2)$, then terminate.
6. Add the fiber link with length $l_N(m_1, m_2)$ to E. Go to step 4.



(a) (b)

Figure 3.4: Italy 116 MC node instance: (a) Cable reference (b) Potential fiber links

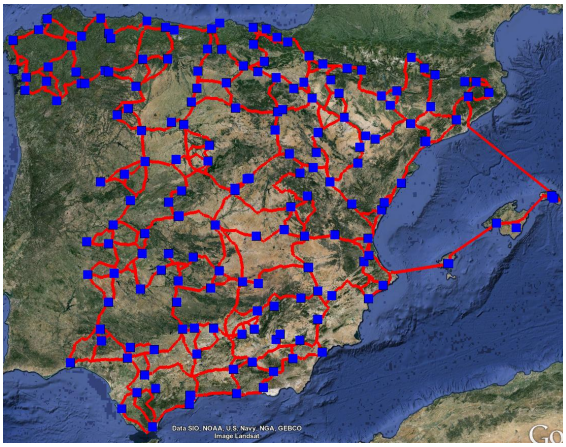
We interpret the resulting graph G as the network of potential fiber links connecting the given set of MC nodes, see Figure 3.3(b), Figure 3.4(b), and Figure 3.5(b). For G it holds that every MC node can reach any other MC node within a distance of at most two times the shortest path distance in the reference network N. Projecting the shortest path realizations of the fiber links E to N we obtain the cable reference network which can be seen in Figure 3.3(a), Figure 3.4(a), and Figure 3.5(a).

We assume that each of the links E may carry an arbitrary number of fibers. The individual number of fiber links for each of the instances is reported in Table 3.2.

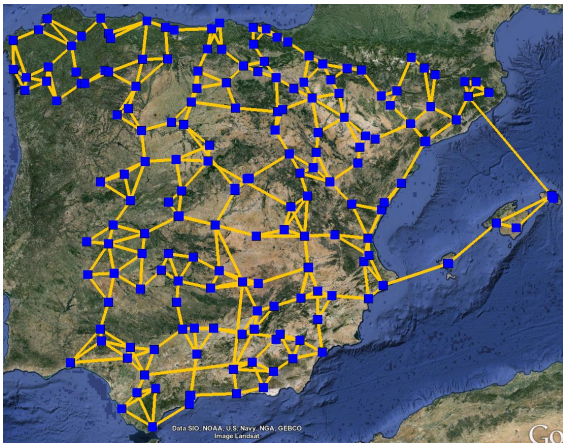
Traffic matrices

We use mainly three traffic scenarios in this section, which we call A, B, and C. These are based on the scenarios defined in Deliverable D2.8 [11]. Scenario A assumes a daily download of about 4.5 GB per user, which is close to the prediction for 2018 of Cisco [33]. Scenarios B and C assume daily downloads of 40 GB and 280 GB, respectively, which can be seen as 2030, 2040 forecasts. For more details on traffic modeling we refer the reader to [6] and [11].

For the mentioned network instances this results in traffic matrices with total traffic volumes reported in Table 3.3. These values depend on the number of households in the respective countries and the number of MC nodes of the particular instance. We refer to [5, 6] on how traffic matrices are computed. Notice that we report the sum of the two directional traffic values between any two nodes. However, for dimensioning we consider the maximum of the two values because all interfaces are considered being bidirectional, see also [6].



(a)



(b)

Figure 3.5: Spain 179 MC node instance: (a) Cable reference (b) Potential fiber links

3.4 Solution methodology

There are several exact and heuristic approaches to tackle multi-layer network design, see for instance [26, 59, 60, 64–66, 73]. The most common approach used in practice is probably based on a bottom-up approach and decomposing the layer coupling. Starting from the highest network layer, each layer is dimensioned individually. Each layer dimensioning introduces the input in terms of demands for the capacity dimensioning of the next layer in the stack. In terms of the introduced DISCUS layers, this means first providing a virtual topology with capacitated links that is able to handle the given traffic matrix (ignoring the physical topology), and, only in a second step, routing the virtual capacities (40G, 100G, 400G channels) in the fiber network, while minimizing the fiber cost.

This approach, however, might yield to sub-optimal solutions, and may cause infeasibilities. In fact, by decomposition it is not guaranteed to find a solution even if there are many, see [72]. Moreover, decomposition ignores the fact that failures may occur in the lowest network layers (in the cable or duct layer) but affect paths and services across all layers.

Starting from a given set of MC nodes M, a corresponding fiber network $G = (M, E)$ as introduced above, and a set of service links $S \subseteq M \times M$ with demand values $d_s > 0, s \in S$

Instance			Traffic in Tbps	
Country	$\max_{Km}$	Demand scenario	Reflected	Core
UK	110 km	A	178	221
UK	110 km	B	378	956
UK	110 km	C	1,373	6,014
Italy	115 km	A	152	194
Italy	115 km	B	323	833
Italy	115 km	C	1,174	5,204
Spain	115 km	A	253	321
Spain	115 km	B	536	1,404
Spain	115 km	C	1,944	8,736

Table 3.3: Total traffic volumes for the different network and demand scenarios. We state the total reflected traffic (traffic with source and target below the MC node not entering the core) and total core traffic.

(the traffic matrix), we will follow an integrated approach that tries to dimension the virtual topology and physical topology simultaneously while minimizing the installation cost.

Following the terminology from [72] and we use a *path-flow over path-flow* model (with *explicit light-paths* [64, 65] or *disaggregated flow* [72]). That is, in both layers, the virtual channel layer, and the physical fiber layer we work with explicit set of paths. We only let the optimization model decide which of the paths to chose. This approach has the flexibility to work with different sets of preselected paths, easily integrating additional constraints on the path realizations such as distance or topology restrictions, see Section 3.2. Of course, since we do not work with column generation, we can speak of optimality only with respect to the chosen set of paths.

Starting from the topology $G = (M, E)$ of all potential fiber links we consider a preselection of paths L in this topology. Each of these paths can be seen as the realization of a virtual link, a potential channel connection. All preselected potential virtual links connecting the MC nodes form a virtual topology $H = (M, L)$, see Figure 3.1.

Given the set of signals T, we denote by T_ℓ the subset of signals that can be used on the optical channel link $\ell \in L$, that is, the length of the realization of ℓ does not exceed the signal reach. Clearly, we may exclude from L all virtual links ℓ with $T_\ell = \emptyset$.

In case of fixed-grid T contains 6 signal types: 40G, 100G, 400G, each either with or without the use of regenerators leading to a maximum signal reach of 5,000 kilometers. Similarly in case of flex-grid we have 12 different signal types, depending on the reach, the bitrate, the required bandwidth slots, and the use of regenerators, see Table 2.2.

We further denote by $\mathcal{P}$ all (virtual) paths in the virtual network $H = (M, L)$. All paths corresponding to a particular service s are denoted by $\mathcal{P}_s$, that is, $\mathcal{P}_s$ contains all virtual paths that can be used to realize the service link. Recall that within the strict optical island concept, any path in $\mathcal{P}$ should have not more than one hop, see 3.2.

We introduce the following three types of variables: Binary variable f_p will indicate whether virtual path $p \in \mathcal{P}$ is used or not. Integer variables y_ℓ^t counts how many optical channels with signal type $t \in T$ are active on the virtual link $\ell \in L$. Eventually, integrals x_e count how many fibers are used on the fiber link e.

Given these variables the following model (TL) optimizes virtual and physical topology simultaneously:

$$(TL) \qquad \min \sum_{\ell \in L} \sum_{t \in T_\ell} \kappa^t y_\ell^t + \sum_{e \in E} \kappa_e x_e \tag{3.1}$$

$$\sum_{p \in \mathcal{P}_s} f_p = 1, \qquad \forall s \in S, \tag{3.2}$$

$$\sum_{s \in S} \sum_{p \in \mathcal{P}_s} d_s f_p - \sum_{t \in T_\ell} c^t y_\ell^t \leq 0, \qquad \forall \ell \in L, \tag{3.3}$$

$$\sum_{\ell \in L: e \in \ell} \sum_{t \in T_\ell} r^t y_\ell^t - B x_e \leq 0, \qquad \forall e \in E \tag{3.4}$$

$$f_p \in \{0,1\} \qquad \forall p \in \mathcal{P}, s \in S$$

$$y_\ell^t, x_e \in \mathbb{Z}_+ \qquad \forall \ell \in L, t \in T_\ell, e \in E$$

Constraints (3.2) guarantee that exactly one path is chosen for each service. In an optical island this path is a direct virtual link between the two MC nodes. The inequality system (3.3) ensures that enough optical channels are used on a virtual link to carry the (packet) flow (in Gbps) of all paths using the link. The term d_s denotes the traffic of the service while c^t is the bitrate capacity of the channel (40G,100G, or 400G). Similarly, system (3.4) ensures enough fibers on all fiber links. In these inequalities, the term r^t refers to the number of bandwidth slots consumed by signal type t and B is the number of slots provided by a fiber (88 or 120). Notice that we ignore wavelength assignment in this model.

Objective (3.1) minimizes the cost of all required resources. The term κ^t denotes the cost of an optical channel of type $t \in T$. It includes the cost of two transponders and, if necessary, the cost of a regenerator. The term κ_e denotes the cost of a fiber. In this case we include the cost for WDM terminals on both sides, the cost for amplification (OLA) and the cost for DGE, see 2.2.

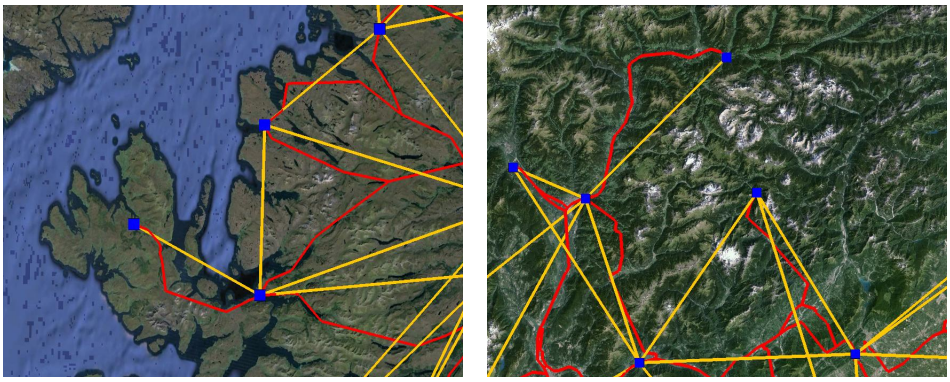
3.4.1 A first step towards resiliency

Model (TL) is very flexible if used with different physical path sets L and virtual path sets $\mathcal{P}$. However, it ignores resiliency in the sense that a failing fiber may cause services to fail. Resiliency within DISCUS is an end-to-end concept that already includes the dual homing from LE to MC sites. It clearly, has to be tested from an end-to-end perspective, see also Chapter 4-6. In this chapter, we are not claiming to get solutions that are resilient down to the level of cables and streets. We show how, in principle, a certain level of survivability can be guaranteed across multiple layers. However, we go down to the fiber layer only (ignoring that two different fibers may follow the same street or duct system) and we do not introduce a hard resiliency constraint. Chapter 4 shows how solutions coming from our optimizations can be improved such as to increase resiliency even down to the level of streets.

To introduce a certain level of resiliency we will use the following routing principle:

1. We route the traffic of each service on two different virtual paths p_1, p_2.
2. We verify that for any two virtual links ℓ_1, ℓ_2 with $\ell_1 \in p_1$ and $\ell_2 \in p_2$, the two physical fiber realizations of ℓ_1 and ℓ_2 do not use a common fiber link.

This guarantees that any single fiber failure in the core does not lead to service interruptions as there is always a second open path with enough capacity between the end-nodes of the service. Notice that this ensures survivability on the fiber layer only. It does not guarantee resiliency on the cable or street level, see Figure 3.6. That is, we implicitly assume that two fibers having a cable/street link in common but take a different path in the cable/street topology are separated on the common link (in different cables, on different sides of the street), also see Chapter 4.



(a) (b)

Figure 3.6: Zoom into the fiber (yellow) and cable (red) topologies of the (a) UK and (b) Italy. We can see that disjointedness on the fiber level does not mean disjointedness on the cable level. We also see one MC in both cases that has only one possible fiber connection to the network.

Moreover, it turns out that for some instances caused by non-sufficient connectivity in the reference network not all MC nodes are connected to the fiber network by two disjoint core fiber routes, see Figure 3.6. At this point we could simply introduce additional links to the fiber layer as these would probably exist in practice. Alternatively, we could remove these MCs from the distribution. However, this might introduce infeasibilities w.r.t the LE-MC connections. We decided to follow a third approach. We relax requirement 2. above, introduce a slack variable, and put into the objective function. That is, instead of a hard constraint 2. we introduce a soft constraint, while minimizing its violation.

The resulting overall survivability enhanced model (TLS) is as follows:

$$(TLS) \qquad \min \sum_{\ell \in L} \sum_{t \in T_\ell} \kappa^t y_\ell^t + \sum_{e \in E} \kappa_e x_e + \sum_{s \in S} z_s \tag{3.5}$$

$$\sum_{p \in \mathcal{P}_s} f_p = 2, \qquad \forall s \in S, \tag{3.6}$$

$$\sum_{s \in S} \sum_{p \in \mathcal{P}_s} d_s f_p - \sum_{t \in T_\ell} c^t y_\ell^t \leq 0, \qquad \forall \ell \in L,$$

$$\sum_{\ell \in L: e \in \ell} \sum_{t \in T_\ell} r^t y_\ell^t - B x_e \leq 0, \qquad \forall e \in E$$

$$\sum_{p \in \mathcal{P}_s: e \in p} f_p \leq 1 + z_s, \quad \forall s \in S, e \in E \tag{3.7}$$

$$f_p, z_s \in \{0,1\} \qquad \forall p \in \mathcal{P}, s \in S \tag{3.8}$$

$$y_\ell^t, x_e \in \mathbb{Z}_+ \qquad \forall \ell \in L, t \in T_\ell, e \in E \tag{3.9}$$

Notice that we have marked changes to model (TL) in blue. Constraints (3.2) have changed to (3.6) requiring two virtual paths per service instead of only one. Constraints (3.7) are the conflict constraints. They forbid paths of the same service to use the same fiber links unless binary slack variable z_s is switched on. In this case at most two paths may use the same fiber link. However, the number of such situations is minimized in the new objective (3.5), where the term $\sum_{s \in S} z_s$ counts how many services do violate the resiliency requirement on at least one fiber link.

3.4.2 Path generation

Before solving models (TL) or (TLS) we have to generate reasonable sets of paths L in the fiber graph $G = (M, E)$ and given the resulting virtual graph $H = (M, L)$ we have to provide a set of virtual paths $\mathcal{P}$. Clearly, all these paths should be short but they should also be designed to share common sources (fibers, optical channels). Moreover, in order to fulfill the resiliency constraint (3.5) (with $z_s = 0$ if possible) we need paths $\mathcal{P}$ that are disjoint when mapped to the fiber graph G.

Recall that in case of the pure optical island concept the set $\mathcal{P}$ is in fact identical to L. more precisely each path in $\mathcal{P}$ consists of exactly one link in L. However, we will compare more general architectures (with optical islands) such that we allow $\mathcal{P}$ to contain paths with inner nodes, allowing to groom traffic at intermediate MCs.

Our path generation process consists of three main steps:

- Path computation
- Path expansion
- Path filtering

Path computation

We may distinguish path computation in the physical and the virtual layer.

In the physical layer, we mainly use the Dijkstra's algorithm [39] to compute short paths and the Suurballe's algorithm [57, 90] to compute short and disjoint paths. In addition we heuristically determine disjoint paths by iteratively computing shortest paths with link weights that forbid the links of the previous iteration. We call these paths *alternative shortest paths*.

In the default settings, for any pair of MC nodes, we compute a shortest path, 3 disjoint paths using Suurballe's algorithm, and 2 alternative shortest paths. All these paths create the set of virtual links L, a set of potentially $6 \cdot |M|(|M|-1)$ elements. Recall that we do not allow virtual links that are too long in the sense that there is no signal type with sufficient reach.

In the now created potential virtual layer $H = (M, L)$ we first create all single-hop paths, that is, all paths that use one of the links in L. We further generate all short two-hop paths between any pair of MC nodes. We might, however limit the allowed intermediate MCs (the grooming locations), see below. Then again we compute all shortest paths in H. These are not necessarily single hop paths as there might be pairs of MC nodes that have no direct virtual link (because of the mentioned signal reach restrictions).

We further compute shortest virtual paths with a weight function that is based on the traffic volume. In this case a virtual link is defined short if there is large demand between its MC end-nodes. Intuitively, we want to create virtual paths that follow the traffic pattern and use links/nodes with large demand. All these paths enter the initial set $\mathcal{P}$

We create many more possible virtual paths by Path expansion:

Path expansion

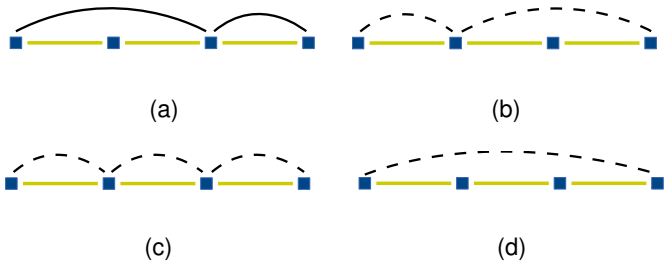


Figure 3.7: Path expansion. (a) Original path; (b)-(d) All possible path expansions. Fiber links are in yellow, virtual links in black.

From each path in $p \in \mathcal{P}$ we create a series of paths by following the same physical path but allowing for different intermediate grooming MCs. This can be done by first mapping the path p to the physical layer G in order to see which MC nodes M_p are visited by p. More precisely, the set M_p contains all inner nodes of p (all grooming locations) and all nodes that are inner nodes of the virtual links contained in p. The latter nodes are optically bypassed by p.

In a second step, we consider all subsets M'_p of M_p and create new paths p' by using all MCs in M'_p as inner nodes following the same physical path, see Figure 3.7. We add all paths p' corresponding to subsets M'_p to $\mathcal{P}$. Clearly, all paths p' use the same fiber links a p, that is, have the same physical representation but they use different virtual links, which allows for different intermediate grooming. Notice that the path expansion operation might create new virtual links that where not generated by the physical path computation above. These are added to L.

Path expansion can be an expensive operation both in terms of CPU-time and memory usage since subsets are generated. To control the necessary computing resources we try to include path filtering mechanisms already in the path expansion.

Path filtering

After path computation and path expansion we use several filtering techniques to reduce the size of the sets $\mathcal{P}$ and L and in order to implement different restrictions on the allowed path realizations. The main filtering criteria are:

- The number of virtual hops $k >= 1$. Paths $p \in \mathcal{P}$ only pass this filter if the number of hops does not exceed k.
- The set of allowed grooming locations $M^{groom} \subseteq M$. Paths $p \in \mathcal{P}$ pass the filtering if they are sing-hop paths or if they contain inner nodes from $M^{groom} \subseteq M$ only.

- We may further force grooming for selected (e.g. small) MC locations $M^{small} \subset M$. If $m \in M^{small}$ then any path in p with m as end-node has to contain a location from M^{groom}. In particular, if both end-nodes are from M^{small} path p needs an intermediate node from M^{groom}. We either set $M^{small} := \emptyset$ or $M^{small} := M \backslash M^{groom}$.

With the first filter, by setting $k := 1$, we force optical island topologies. By combining the last two criteria we may force two-tier topologies, that is, topologies where the traffic from preselected (smaller) nodes needs to get aggregated at preselected inner core nodes.

Clearly, all of the filtering criteria can partially be incorporated already in the path computation and expansion.

Instance				Cost in Mio €			
Country	$\max_{Km}$	Traffic	Scenario	Channel	Fiber	Switch	Total
uk	110 km	A	OPT	331	51	140	522
uk	110 km	A	GROOM	340	46	154	540
uk	110 km	A	ISLAND	475	93	148	716
italy	115 km	A	OPT	494	107	189	790
italy	115 km	A	GROOM	526	111	202	839
italy	115 km	A	ISLAND	769	230	226	1224
spain	115 km	A	OPT	833	182	311	1325
spain	115 km	A	GROOM	871	190	326	1387
spain	115 km	A	ISLAND	1784	504	519	2807

Table 3.4: Cost of the core network for different architectures and networks. Traffic scenario A. Channel cost includes cost for transponders (fixed grid), regenerators, and transceivers. Fiber cost means cost for WDM terminals, OLAs, and DGEs. Switch cost is cost for MPLS-TP switches, Polatis switches as well as for line-cards.

3.5 Computations

In this section, we use the presented data and methodology to dimension the core of the DISCUS architecture. In the first part, we evaluate the cost of optical islands for the network instances from Table 3.2 and compare this cost with two alternative architectures. In the second part we will study how the different architectures scale with increasing traffic volumes. In particular, we will try to understand under which conditions optical islands are cost-effective compared to architectures that allow for traffic grooming in the core or use a second level of traffic aggregation.

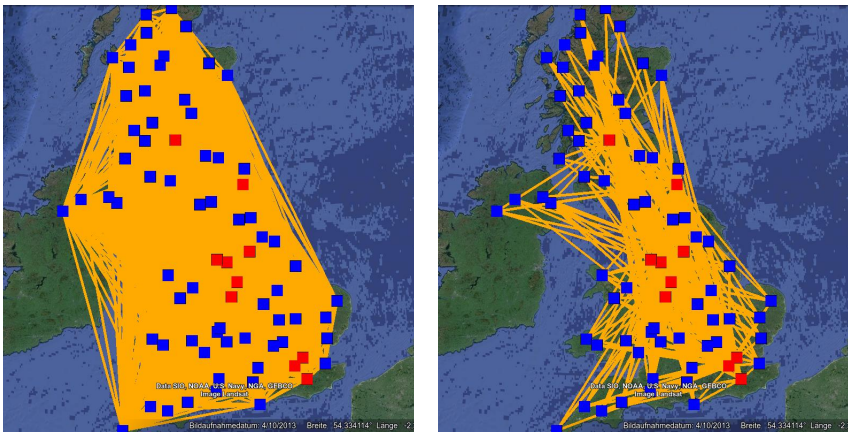
To solve the models (TL) and (TLS) we use the solver CPLEX 12.6 multi-threaded with up to 6 cores and up to 50 GB RAM on a machine with 10 CPUs (2.8 Ghz). We set the time limit to 10,000 seconds and the emphasis of the solver to finding feasible solutions.

Table 3.4 summarizes the results of our first experiment using the traffic scenario A (2018) and model (TLS). We distinguish three different runs of the optimization routine with different settings w.r.t the envisaged architecture:

- ISLAND: We force an optical island architecture by setting $k := 1$ in the path filtering as explained above. Regenerators are used if necessary.

- GROOM: We force a two-tier type architecture by setting M^{groom} to the 10 biggest MC locations. All traffic has to be groomed at these nodes ($M^{small} := M \backslash M^{groom}$). The maximum number of virtual hops is restricted to $k := 3$.
- OPT: We still set M^{groom} to the 10 biggest locations. Grooming is allowed only at these nodes. However, we do not force grooming ($M^{small} := \emptyset$) as in GROOM. The maximum number of virtual hops is restricted to $k := 3$.

Notice that the solution space corresponding to the third setting includes the solution spaces of the first two.



(a) (b)

Figure 3.8: Virtual layer of (a) ISLAND: Full mesh optical island solution (b) GROOM: Solution with grooming at the 10 biggest MCs (in red); all optical channels start or end at one of these 10 nodes.

Table 3.4 shows the cost in Mio € of the different solutions. We state the cost occurring in the core only. That is, for dimensioning the line-cards and switches, we ignore any port needed in the access and we ignore traffic that is reflected at the MC node. It can be seen that a two-tier architecture (GROOM) outperforms the optical island (ISLAND) in terms of cost using traffic scenario A. The cost can be decreased even further if we relax the requirement to groom at one of the 10 big locations and allow for direct links between some of the smaller locations. That is, a mixture of the optical island and the two-tier concept gives the cheapest network. However, the difference between the best cost and the cost for the optical island decreases with a decreasing number of MC nodes. We save 1,482 Mio € (52.8%) for Spain 434 Mio € (35.5%) for Italy, but only 194 Mio € (27.1%) for the UK.

We note that all optical island solutions can be considered being optimal (optimality gaps below 1%). However, in particular the solutions for Italy and Spain and scenarios OPT and GROOM have potential to be improved (optimality gaps larger than 20%). This means that the mentioned differences might in fact be even larger. Moreover, even scenario OPT introduces restrictions on the allowed virtual topologies. Further relaxing these restrictions (allowing for multi-hop paths with $k > 3$, allowing for more grooming locations than 10, etc.) might further decrease the cost for OPT.

This seems to be bad news for the optical island concept unless we have small countries with a possibly small number of MC nodes as in the case of the UK. However, the picture

Country	Instance $\max_{Km}$	Traffic	Scenario	Cost in Mio € Channel	Fiber	Switch	Total
uk	110 km	0.01·A	opt	19	16	23	58
uk	110 km	0.01·A	groom	19	16	23	58
uk	110 km	0.01·A	island	284	74	77	435
uk	110 km	0.1·A	opt	56	20	35	111
uk	110 km	0.1·A	groom	56	20	35	111
uk	110 km	0.1·A	island	286	74	79	438
uk	110 km	0.3·A	opt	143	33	64	240
uk	110 km	0.3·A	groom	162	34	73	269
uk	110 km	0.3·A	island	311	77	89	477
uk	110 km	0.5·A	opt	197	36	88	321
uk	110 km	0.5·A	groom	224	40	99	363
uk	110 km	0.5·A	island	358	82	105	546
uk	110 km	A	opt	331	51	140	522
uk	110 km	A	groom	340	46	154	540
uk	110 km	A	island	475	93	148	716
uk	110 km	2.0·A	opt	591	78	235	904
uk	110 km	2.0·A	groom	655	73	297	1024
uk	110 km	2.0·A	island	721	122	236	1079
uk	110 km	3.0·A	opt	877	114	349	1339
uk	110 km	3.0·A	groom	954	97	431	1483
uk	110 km	3.0·A	island	961	146	329	1435
uk	110 km	4.0·A	opt	1096	132	423	1652
uk	110 km	4.0·A	groom	1202	113	546	1861
uk	110 km	4.0·A	island	1209	175	421	1805
uk	110 km	B	opt	1348	172	508	2028
uk	110 km	B	groom	1510	143	680	2332
uk	110 km	B	island	1445	199	503	2146
uk	110 km	8.0·A	opt	2083	234	789	3105
uk	110 km	8.0·A	groom	2343	200	1065	3608
uk	110 km	8.0·A	island	2189	265	796	3250
uk	110 km	C	opt	8748	959	3217	12923
uk	110 km	C	groom	10328	860	4655	15843
uk	110 km	C	island	8890	1022	3215	13128

Table 3.5: Cost of the core network for the UK network with 75 MC nodes and different traffic scenarios. We used the traffic scenarios A, B, and C and also scaled scenario A with factors from $\{0.01, 0.1, 0.3, 0.5, 2.0, 3.0, 4.0, 8.0\}$.

changes completely if we change the size of the traffic matrix focusing on the scalability of the architectures in terms of future traffic.

In Table 3.5 we report on the same computations (UK network) but with different (scaled) traffic matrices. We use matrix A as in Table 3.4 scaled with a factor from $\{0.01, 0.1, 0.3, 0.5, 1.0, 2.0, 3.0, 4.0, 5.0\}$ and matrices B, C. It can be seen that for smaller traffic volumes an architecture based on grooming traffic clearly outperforms any optical island concept. With traffic matrix $0.01 \cdot A$ the cost of an optical island is 7.5 times the cost for a two-tier architecture. In fact with very small traffic, the cost of an optical island is independent of the traffic as we have to open one channel for each pair of nodes anyway. In case of the UK the total optical island cost is relatively constant between 435 Mio € and 477 Mio € although the traffic increases by a factor of 300.

However with larger traffic volumes the advantage of grooming decreases and at some point optical islands outperform architectures which force to groom. Already a matrix of size 3 times A (around 700 Tbps total core traffic) makes the optical island cheaper than

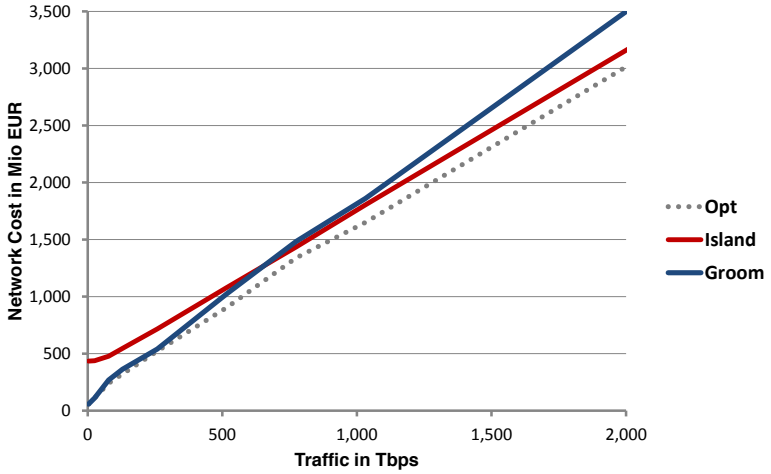


Figure 3.9: Cost of different architectures as a function of the traffic volume, UK network with 75 MC nodes, LEs connected with a maximum distance of 110 km.

the grooming architecture (1435 Mio € versus 1483 Mio €). Figure 3.9 shows the cost curves as a function of the traffic and clearly visualizes the threshold of around 700 Tbps.

Of course, a mixture of both concepts (grooming to fill up the channels and optical islands for scalability), that is scenario OPT, gives the best results as it allows to groom traffic when necessary, that is, for very small MC nodes and for traffic values that are not exact multiples of 40, 100, 400 Gbps. It should also be noticed that for higher volumes of traffic the value found for OPT converges to that for ISLAND, clearly indicating that the use of optical islands tends to be the solution of choice for increasing traffic volumes.

We close with Figure 3.10, which presents similar results for the Italian network. We can see a similar behavior of the cost curves. However, the cost values for GROOM (and also OPT) are not as resilient as for the UK because our optimization terminated with relatively large optimality gaps ($> 20\%$). They can only be seen as an upper bound to the actual cost such that the actual cost function for GROOM (and OPT) might (!) be below the one in Figure 3.10.

3.6 Conclusion

In this chapter, we showed how to solve multi-layer core network design problems in order to assess the impact of the DISCUS optical island concept. For different European countries and different metro-core nodes locations, we solve an integrated planning problem including the dimensioning of the virtual channel layer and the physical cable/fiber layer.

Using our methodology we optimized different types of core networks based on different architecture assumptions. We were able to show that optical islands outperform architectures based on aggregating (grooming) traffic towards an inner core once the traffic volume exceeds a certain threshold depending on the cost model, the number of metro-core nodes and the available channel capacities (40G, 100G, 400G).

For the UK this threshold is around 700 Tbps total core traffic using the cost model from Chapter 2 and the smallest channel bitrate being 40 Gbps.

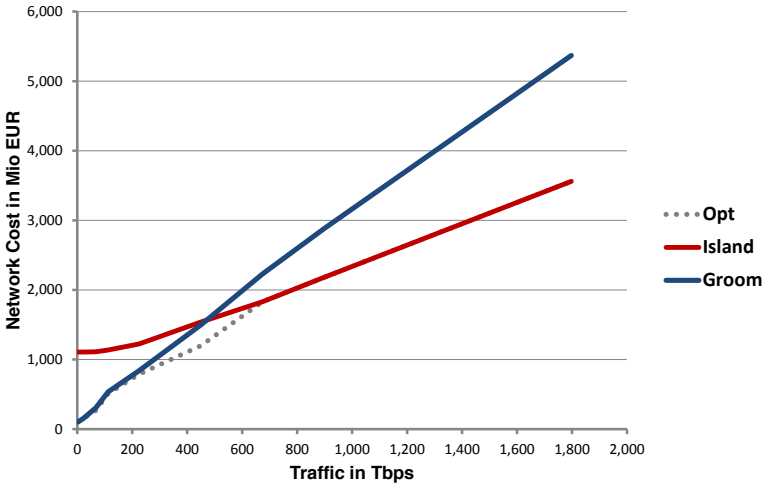


Figure 3.10: Cost of different architectures as a function of the traffic volume, Italy network with 116 MC nodes, LEs connected with a maximum distance of 115 km. The curves of scenarios OPT and GROOM can only be seen as an upper bound to actual cost as our optimization terminated with relatively large optimality gaps.

Chapter 4

Resilient core network planning

Ensuring network survivability in the presence of failures is a crucial prerequisite in providing highly reliable network operation with low outage times. Optical core networks are vulnerable to a wide set of failures, ranging from individual component faults due to physical damage (e.g., link cuts) or fatigue, over disaster-like events affecting entire M/C nodes or wider geographical areas, to intentional malicious activities aimed at disrupting the service. This chapter focuses on ways of providing survivability in the network planning phase by considering single link- and node-component failures as well as deliberate physical-layer attacks targeting service disruption.

Section 4.1 addresses the requirement for survivability from link and/or node failures in the network design phase. Link cuts due to, e.g., construction equipment ploughing through the ducts are the most common cause of component failures. Failures of entire nodes can be seen as disasterous events disconnecting thousands of users. To adress this issue new, 3-step network design approach is proposed which increases the number of DISCUS reference network node pairs for which physically disjoint paths can be found, such that the total distance between all node pairs is minimized and the path lengths do not exceed the optical signal reach.

Section 4.2 focuses on individual node component failures and investigates means of increasing network reliability through the deployment of synthetic programmable ROADMs implemented by Architecture on Demand (AoD). Such nodes provide unprecedented levels of flexibility and are capable of so-called self healing, allowing for node components failures to be healed at the node level without triggering network-level recovery mechanisms. A connection routing approach is proposed to enhance the self-healing functionality and reduce the number of failures which require recovery at the network level.

Section 4.3 studies vulnerability of transparent optical networks to deliberate physical-layer attacks aimed at service disruption. Since such attacks do not occur often, but can cause major wide-area damage in case they do appear, high resource-efficiency is particularly desirable in resiliency schemes which consider attacks. Therefore, consideration of attacks needs to be incorporated into the network planning procedures as an additional damage-reduction criterion while maintaining standard optimization objectives typically aimed at minimizing resource usage and cost. To this end, an attack-aware approach for dedicated path protection is developed to reduce the number of connection which remain unprotected in the presence of an attack while using the same number of wavelengths as standard, resource-minimizing protection approach.

Section 4.4 focuses on advantages of dual-homed networks in providing resource-efficient survivability from core link failures. Dual homing has the potential to improve resource-usage efficiency by allowing greater flexibility in the selection of backup paths. A dedicated path protection approach is proposed to utilize this property and satisfy the availability requirements of connection requests at a lower resource consumption.

4.1 Resilient core network design

Survivability to device failures is a fundamental requirement in core networks since a single failure in the network could interrupt on-line services in social and business life [71]. Finding a minimum distance resilient optical core network is an intractable problem since it includes the constrained shortest path problem as special case [99]. Even the network design is fixed, for general undirected graphs, finding node-disjoint paths is NP-complete [31].

The problem considered in this section is finding the minimum distance design (MDD) networks for both abstract and street levels where the disjointness is maximised. More formally, a set of trails from given street network with minimal total length are selected so that there exists two bounded paths between each pair of metro core nodes with maximal disjointness. In addition to the theoretical computational complexity of the problem, due to the size of the nation-wide street networks where nodes and edges refer road junctions and trails, there is also a scalability problem in practice. Therefore, in this section, we decompose the problem into three steps.

Developed 3-step approach starts with abstraction process which reduces the street network with road junctions and road links (i.e., trails) to an abstract network where nodes and edges refer to the metro-core nodes and the links between them. After that, in the second step of the algorithm, a mixed integer programming (MIP) decomposition based algorithm is used to find the minimum distance bounded node disjoint resilient abstract network. Although it is possible to find high level of node disjointness after the second step in the abstract level, the actual disjointness which is computed after projecting/mapping the abstract solution into the street network is most likely less than the desired level. Hence, in the last step of the algorithm, we propose a greedy method to improve the disjointness of the paths between metro-core nodes. A flowchart of the 3-step algorithm is presented in the following and details of each step are discussed in the next sections.

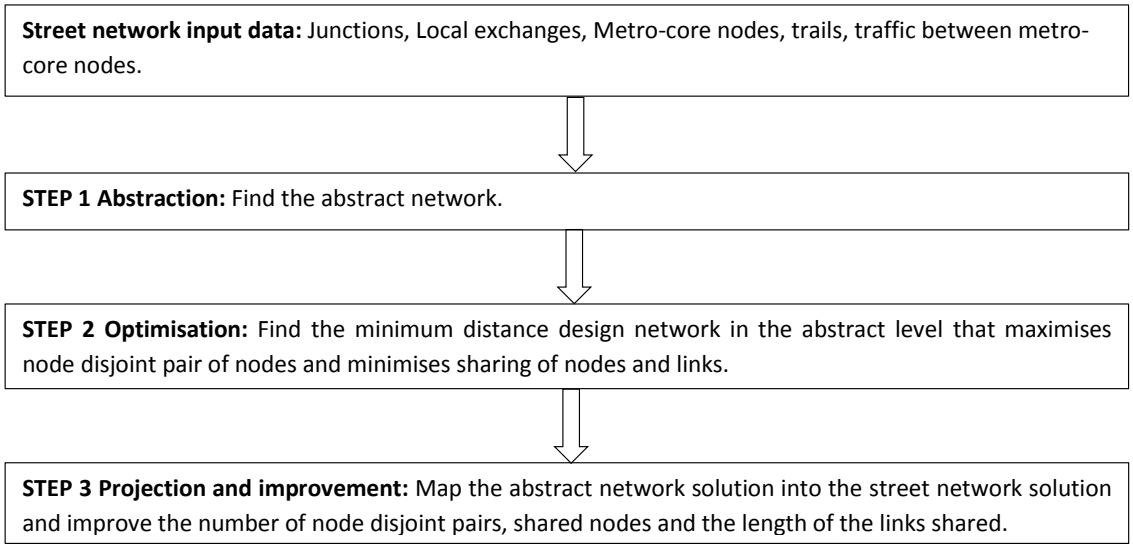


Figure 4.1: Flowchart of the 3-step algorithm for resilient core network design

4.1.1 STEP 1: Abstraction

In this first step, we reduce the street network with road junctions and trails into an abstract network where metro-core nodes are nodes and the possible links between them are edges of the graph. Details of the abstraction procedure is given in the following.

- Let $G^r = \langle V^r, E^r \rangle$ be the graph corresponding to the street network.
- Let $G^a = \langle V^a, E^a \rangle$ be the graph corresponding to the abstract network. Here V^a correspond to the set of MC-nodes.
- $SP(i, j, G^r - B)$ is the shortest path between metro core nodes i and j in G^r which excludes all other metro-core nodes to guarantee that if it exists, it does not rely on any other metro-node. If such a path does not exist, initially it is known that there is no node-disjoint path between the corresponding metro-nodes.

Based on these definitions, the abstraction procedure is given in the following which is executed for every pair of metro-core nodes and computes the length of the abstract link between these pairs as the length of the shortest path in the street network.

---

for all $\{\langle i, j \rangle | i \in N \wedge j \in N \wedge i \neq j\}$ **do**
$B \leftarrow N \setminus \{i, j\}$
$|e| = |SP(i, j, G^r - B)|$

Size of the resulting abstract UK, Italy and Spain networks are summarized in Table 4.1. Note that sizes of the initial abstract and street networks refer to the sum of the all possible links.

Due to overlapped trails in different abstract network links, size of the sreet network is much smaller than the abstract network. Assume that it is required to construct a shortest path between metro core nodes 31 and 95 in the abstract level. And furthermore, first two links in the shortest path are the abstract links between metro core nodes (31-32) and (32-84). When we compute the projection of abstract link (31-32) in the street network, we observe that the last two trails of this abstract link are (in order of occurrence in the path) trail number 29237 and 15186. Similarly, when we compute the projection of abstract link (32-84) in the reference network, first two trails are (in the order of the occurrence), trail numbers 15186 and 29237. In other words, the same trails are used in the path between 31-32 and 32-84 for connecting metro core node 32 to others. Since the length of the abstract links are immutable, we are incurring the length of these two trails twice in the abstract level solution. However, when we compute the projection in the street network level, we're omitting the repeated occurrences of the same trails. Hence, the total length of the shortest path between 31 and 95 is in the abstract network is longer than its projection in the street network. Therefore, the total length of the links (i.e., size of the network) in the abstract level is more than its mapping in the street level.

In addition, an initial analysis on these network topologies also show that some pair of nodes don't have node disjoint paths. There are 8 pair of metro-core nodes related with Sardinia, Sicily and Elba islands for Italy topology which have no node-disjoint paths in the street network withing the signal reach. Similarly, in UK reference network, metro core node MC—scotland—Ex—4587 requires a mandatory link to metro-core node MC—scotland—Ex—4754 to connect to the nationwide UK network. Therefore, any k paths originating from or terminating in MC—scotland—Ex—4587 are not node disjoint.

	UK	Italy	Spain
Number of metro core nodes (N)	75	116	179
Number of nodes in the street network, road junctions, (V^r)	15608	23689	18819
Number of edges in the street network, trails, (E^r)	23027	32702	26480
Number of nodes in the abstract network (V^a)	75	116	179
Number of edges in the abstract network (E^a)	2775	6670	15931
Number of not disjoint pair of metro-nodes in the street network	74	8	0
Number of not disjoint pair of metro-nodes in the abstract network network	74	8	0
Initial abstract network size, STEP 1 (km)	1279529	4835343	8765621
Initial street network size, STEP 1 (km)	22880	23155	37606

Table 4.1: Size of the Street and the Initial Abstract Networks

4.1.2 STEP 2: Designing Minimum Distance Node Disjoint Abstract Network

Once we create the abstract graph which consists of metro-core nodes, N, and all possible links between them, L, in this step, we select a subset of these links to minimise the total length and to ensure maximal node disjointness between any pair of nodes. Formal definition of the problem solved in this step is given in the following :

Definition 1 (All-Pairs k Node-Disjoint Length-Bounded Paths Problem)**.** *Given a graph $\langle N, L\rangle$ where N is a set of nodes N and L is a set of links, the All-Pairs k Node-Disjoint Length-Bounded Paths Problem (AkNLPP) is to determine $k \geq 2$ bounded paths between each pair of nodes $\{i, j\} \subseteq N$. The objective is to first minimise the number of pairs of nodes whose k paths are not node-disjoint, then minimise the sum of the number of nodes shared between two or more selected paths associated with the pairs of nodes which are not node-disjoint, then minimise the sum of the lengths of the links shared between two or more selected path of pair of nodes that are not node-disjoint, then minimise the weighted sum of the selected links used by all the selected paths and finally minimise the weighted sum of the lengths of all the paths.*

There are five cases for a given pair of MC-nodes that can be observed while solving resilient network design problem:

1. There exists k disjoint paths and length of each one of them is within the maximum signal reach.
2. k disjoint paths do not exist and the length of the shortest path is within maximum signal reach.
3. k disjoint paths do not exist and the length of the shortest path is more than the maximum signal reach.
4. There exists k disjoint paths but the length of only some of them is within maximum signal reach and the length of the others exceeds the reach.

5. There exists k disjoint paths but the length of each one of them exceeds the maximum signal reach.

However, even classifying a given pair of nodes into one of these options is also an NP-hard problem. Hence, there is a need for effective heuristic methods for solving the AkNLPP problem. Beforehand, in the following, we provide a novel model formulation to mathematically define the problem with the following multi-objectives in the order of preferences

1. Maximise the number of pairs with 100% node disjoint paths which are classified as case 1.
2. Maximise partial disjointness which means minimising the number of nodes and the length of the links shared between paths if 100% node disjoint paths are not possible which are classified in cases 2– 5.
3. Minimise the distance based size of the network.
4. Minimise the length of the paths due to minimise transponder costs.

where the following constraints must be ensured

1. Find k paths for each pair of metro-nodes $\tau \in \dagger$.
2. Length of each of k paths must be less than the maximum signal reach.

The reason why we distinguish between the number of pairs with fully disjoint paths (i.e., no shared nodes between paths, 100% disjointness) and the partial disjointness (i.e., at least one node and/or link is common between paths) is due to model five special cases described above. In other words, any method that aims to solve "All-Pairs k Node-Disjoint Length-Bounded Paths Problem" must search for finding fully k disjoint paths for pair of metro-nodes and for ones that it is not possible, must find k paths that shares minimum number of nodes and length of links common. Finally, the same level of disjointness must be ensured with a network with minimum size in terms total length of the links and the minimum sum of path lengths. In the following, we provide a general mathematical formulation to model "All-Pairs k Node-Disjoint Length-Bounded Paths Problem".

Sets and Indices

- $i \in N$: the set of metro-core nodes,
- $l \in L$: the set of all all possible links between metro-core nodes and is also denoted as $\{i, j\}$
- $e, a \in E$: set of all directed edges between metro-core nodes. A link between metro-nodes i and j is undirected. If a physical link is installed between any pair of metro-core nodes, say $i \in N$ and $j \in N$, then two edges $\langle i, j\rangle$ and $\langle j, i\rangle$ are defined because of bi-directional data transfer.
- $\tau \in \dagger = \{\langle i, j\rangle | i \in N \wedge j \in N \wedge i \neq j\}$: the set of all directed pairs of metro-nodes between which connection requests can be made

- $k \in K$: index of node disjoint paths

Parameters

- c_l: the cost of a link $l \in L$ end is equal to the length of the link.
- $|e|$: length of a directed edge $e \in E$ which is equal to the length of the corresponding link.
- $\texttt{In}(i)$ ($\texttt{Out}(i)$): all edges entering (leaving) node $i \in N$.
- λ : the maximum reach of any allowed optical signal, which is 2430km.
- $s(\tau)$ ($t(\tau)$) : source and target nodes of pair $\tau \in \dagger$ respectively.
- ε_τ : traffic between source and target nodes of pair $\tau \in \dagger$.
- α, β, γ and θ: big numbers that are used to model preference of objective functions such that $\alpha \gg \beta \gg \gamma \gg \theta$.

Variables

- w_τ : a binary variable for each pair of metro nodes that is true if and only if any two of the k paths share at least one node. In other words, this binary variable is true if ad only if all of k paths are not fully disjoint (i.e., partially disjoint).
- $z_{\tau i}$: a binary variable for each pair and node that is true if and only if node $i \in N$ is shared by at least two of $k \in K$ paths of pair $\tau \in \dagger$.
- y_l : a binary variable for each link $l \in L$ which is true if and only if the link is included in the resilient optical island.
- $x^k_{\tau e}$: a binary variable that is true if and only if edge e is used by path k of directed pair of nodes $\tau \in \dagger$.
- $q_{\tau e}$: a binary variable that is true if and only if edge e is is shared by at least two of $k \in K$ paths of pair $\tau \in \dagger$.

All-Pairs k Node-Disjoint Length-Bounded Paths Problem

$$\min \alpha \times \sum_{\tau \in \dagger} w_\tau + \beta \times \sum_{\tau \in \dagger} \sum_{i \in N} z_{\tau i} + \gamma \times \sum_{\tau \in \dagger} \sum_{e \in E} |e| \times q_{\tau e} + \theta \times \sum_{l \in L} c_l \times y_l + \sum_{\tau \in \dagger} \varepsilon_\tau \times \sum_{e \in E} \sum_{k \in K} |e| \times x^k_{\tau e} \tag{4.1}$$

$$\forall_{\tau \in \dagger} \forall_{k \in K} : \sum_{e \in \texttt{In}(s(\tau))} x^k_{\tau e} = 0, \sum_{e \in \texttt{Out}(s(\tau))} x^k_{\tau e} = 1 \tag{4.2}$$

$$\forall_{\tau \in \dagger} \forall_{k \in K} : \sum_{e \in \texttt{Out}(t(\tau))} x^k_{\tau e} = 0, \sum_{e \in \texttt{In}(t(\tau))} x^k_{\tau e} = 1 \tag{4.3}$$

$$\forall_{\tau \in \dagger} \forall_{i \in N \setminus \{s(\tau), t(\tau)\}} \forall_{k \in K} : \sum_{e \in \texttt{In}(i)} x^k_{\tau e} = \sum_{e \in \texttt{Out}(i)} x^k_{\tau e} \tag{4.4}$$

$$\forall_{\tau\in\dagger}\forall_{k\in K}:\sum_{e\in E}|e|\times x^k_{\tau e}\leq\lambda \tag{4.5}$$

$$\forall_{\tau\in\dagger}\forall_{l\equiv\{i,j\}\in L}\forall_{e\in\{\langle i,j\rangle\cup\langle j,i\rangle\}}\forall_{k\in K}:y_l\geq x^k_{\tau e} \tag{4.6}$$

$$\forall_{\tau\in\dagger}\forall_{i\in N\setminus\{s(\tau),t(\tau)\}}:\sum_{e\in\mathtt{In}(i),k\in K}x^k_{\tau e}\leq 1+(|K|-1)\times z_{\tau i} \tag{4.7}$$

$$\forall_{\tau\in\dagger}:\sum_{i\in N}z_{\tau i}\leq|N|\times w_\tau \tag{4.8}$$

$$\forall_{\tau\in\dagger}\forall_{e,a\in E|s(e)<t(e)\wedge s(e)=t(a)\wedge t(e)=s(a)}:\sum_{k\in K}x^k_{\tau e}+x^k_{\tau a}\leq 1+(|K|-1)\times q_{\tau e} \tag{4.9}$$

$$\forall_{\tau\in\dagger}\forall_{i\in N}:\sum_{e\in E,k\in K|s(e)=i\wedge t(e)=i}x^k_{\tau e}\geq z_{\tau i} \tag{4.10}$$

$$\forall_{\tau\in\dagger}\forall_{e\in E|s(e)<t(e)}:\sum_{k\in k}(x^k_{\tau e}+\sum_{a\in E|s(a)=t(e)\vee t(a)=s(e)}x^k_{\tau a})\geq q_{\tau e} \tag{4.11}$$

$$\forall_{\tau\in\dagger}:w_\tau\leq\sum_{i\in N}z_{\tau i} \tag{4.12}$$

$$\sum_{l\in L}y_l\geq|N| \tag{4.13}$$

$$\forall_{\tau\in\dagger}:w_\tau\in\{0,1\} \tag{4.14}$$

$$\forall_{\tau\in\dagger}\forall_{i\in N}:z_{\tau i}\in\{0,1\} \tag{4.15}$$

$$\forall_{\tau\in\dagger}\forall_{e\in E}:q_{\tau e}\in\{0,1\} \tag{4.16}$$

$$\forall_{\tau\in\dagger}\forall_{e\in E}\forall_{k\in K}:x^k_{\tau e}\in\{0,1\} \tag{4.17}$$

$$\forall_{l\in L}:y_l\in\{0,1\} \tag{4.18}$$

Objective function (4.1) minimises the weighted sum of the number of partially disjoint pairs, total number of shared nodes, weighted sum of the lengths of the shared lengths for partial disjoint pairs, the total length of the links used and the total length of the paths for each pair of nodes that are weighted by the traffic between them. Note that we use weighted-sum method [79] to formulate this multi-criteria objective function (4.1).

Constraints (4.2) - (4.6) are used to model the network design part of the problem. Constraints (4.2) and (4.3) ensure that for each pair of nodes, no route should reach the source node or leave the target node and exactly one edge must leave (reach) from (to) the source

(the target) node. Constraints (4.4) enforces that at each intermediate node of paths the incoming degree should be equal to the outgoing degree. The length of the route from source to target cannot exceed the threshold for each path of $\tau \in \dagger$ with constraints (4.5). Constraints (4.6) guarantee that a link l between nodes i and j is selected if any corresponding directed edge between them is used by at least one of the k paths of any pair of nodes.

Constraints (4.7) - 4.9 model node and link sharing. Constraints (4.7) label common intermediate nodes for each $\tau \in \dagger$ if they are shared by at least two of k paths. Constraints (4.8) indicate that paths of pair of metro-node $\tau \in \dagger$ are not node disjoint if at least one node is shared between two or more paths. Similarly, 4.9 label common links that are shared between two or more paths of each $\tau \in \dagger$.

Constraints (4.10) - (4.12) are formulated as valid inequalities which help to prune the search space. Constraints (4.10) bound the value of z variables for each pair and node and guarantee that the variable is equal to 1 if and only if that node is used in more than one path of the corresponding pair of nodes. Similarly, Constraints (4.11) ensure that a link is labeled as ”shared” for a pair of nodes if and only if the link is shared by at least two of k paths of that pair. Constraints (4.12) give a valid upper-bound for node disjoint indicator variables for each pair of nodes $\tau \in \dagger$ and ensure that w_τ variable take value 1 if and only if there exist at most one common node shared by some of k paths of that pair. Total number of selected link is at least equal to the number of nodes which is a valid inequality for the resilient optical network design problem and formulated as in (4.13).

The MIP model provided above is able to capture all properties of the problem and aims to solve the problem at once with all pair of nodes. However, solving the model is intractable especially for nation-wide networks where the number of pair of metro-core nodes might be more than 10.000.Therefore, we develop a decomposition algorithm (Algorithm 1) which solves the optimisation model for only one pair of nodes at a time and updates the set of used links iteratively.

Algorithm 1 also decomposes the complex objective function in equation 4.1 into two parts by using two different MIP models called in methods FINDCHEAPESTBOUNDEDFULLYNODEDISJOINTPATHS and FINDCHEAPESTBOUNDEDMAXIMALNODEDISJOINTPATHS. For a given pair of metro core node τ, Algorithm 1 first tries to find K node disjoint paths while minimising the distance based cost by calling the FINDCHEAPESTBOUNDEDFULLYNODEDISJOINTPATHS method. If the MIP model used in FINDCHEAPESTBOUNDEDFULLYNODEDISJOINTPATHS method can't find K node disjoint paths, then FINDCHEAPESTBOUNDEDFULLYNODEDISJOINTPATHS method is called and K paths with maximal disjointness are found. Details of these methods are given in Figures 4.2 and 4.3. Note that constraints 4.25 in Figure 4.2 are formulated to forbid direct links between the source and target nodes.

Algorithm 1 BOUNDEDNODEDISJOINTNETWORK($\dagger$,N,L,K)

Sort $\dagger$ *based on non-decreasing order of* |SHORTESTPATH(τ, L)|
$E \leftarrow \{\langle i,j\rangle \cup \langle j,i\rangle | \{i,j\} \in L\}$
$\forall_{\tau\in\dagger}\ \forall_{k\in K}\ \forall_{e\in E}\ x^k_{\tau e}$ is a binary variable
$L^b \leftarrow \emptyset$, ${path^K}_\tau \leftarrow \emptyset$
while $\dagger \neq \emptyset$ **do**
  Remove the first element τ from the sorted set $\dagger$
  ${x^K}_\tau \leftarrow$FINDCHEAPESTBOUNDEDFULLYNODEDISJOINTPATHS(τ,N,L,L^b,K)
  if ${x^K}_\tau = \emptyset$ **then**
    ${x^K}_\tau \leftarrow$FINDCHEAPESTBOUNDEDMAXIMALNODEDISJOINTPATHS(τ,N,L,L^b,K)
  ${path^K}_\tau \leftarrow \{e \,|\, x^k_{\tau e} = 1 \wedge e \in E\}$
  $L^b \leftarrow \{l \equiv \{i,j\} \,|\, x^k_{\tau e} = 1 \wedge e \equiv \langle i,j\rangle \in E\}$
return $[L^b, path^K]$

$$\min \alpha \times \sum_{l\equiv\{i,j\}\in L\setminus L^b} \sum_{e\in\{\langle i,j\rangle\cup\langle j,i\rangle\}} \sum_{k\in K} c_l \cdot x^k_{\tau e} + \varepsilon_\tau \times \sum_{e\in E}\sum_{k\in K} |e| \cdot x^k_{\tau e} \tag{4.19}$$

$$\forall_{k\in K}: \sum_{e\in \mathtt{In}(s(\tau))} x^k_{\tau e} = 0, \sum_{e\in \mathtt{Out}(s(\tau))} x^k_{\tau e} = 1 \tag{4.20}$$

$$\forall_{k\in K}: \sum_{e\in \mathtt{Out}(t(\tau))} x^k_{\tau e} = 0, \sum_{e\in \mathtt{In}(t(\tau))} x^k_{\tau e} = 1 \tag{4.21}$$

$$\forall_{i\in N\setminus\{s(\tau),t(\tau)\}}\forall_{k\in K}: \sum_{e\in \mathtt{In}(i)} x^k_{\tau e} = \sum_{e\in \mathtt{Out}(i)} x^k_{\tau e} \tag{4.22}$$

$$\forall_{k\in K}: \sum_{e\in E} |e| \times x^k_{\tau e} \leq \lambda \tag{4.23}$$

$$\forall_{i\in N\setminus\{s(\tau),t(\tau)\}}: \sum_{e\in \mathtt{In}(i),k\in K} x^k_{\tau e} \leq 1 \tag{4.24}$$

$$\sum_{e\in \mathtt{In}(i),k\in K|s(e)=s(\tau)\wedge t(e)=t(\tau)} x^k_{\tau e} \leq 1 \tag{4.25}$$

$$\forall_{e\in E}\forall_{k\in K}: x^k_{\tau e} \in \{0,1\} \tag{4.26}$$

Figure 4.2: FINDCHEAPESTBOUNDEDFULLYNODEDISJOINTPATHS(τ,N,L,L^b,K): FIND K CHEAPEST BOUNDED NODE DISJOINT PATHS BETWEEN THE MC NODES OF τ

Size of the initial networks presented in STEP 1 are reduced as shown in 4.2 by using Algoritm 1 for $K = 2$. Note that MDD (Minimum Distance Design) is used to underline the size of the network after minimum number and total length of the links are selected.

$$\min \beta \times \sum_{i \in N} z_{\tau i} + \gamma \times \sum_{e \in E} |e| \times q_{\tau e} + \alpha \times \sum_{l \equiv \{i,j\} \in L \setminus L^b} \sum_{e \in \{\langle i,j \rangle \cup \langle j,i \rangle\}} \sum_{k \in K} c_l \cdot x^k_{\tau e} + \varepsilon_\tau \times \sum_{e \in E} \sum_{k \in K} |e| \cdot x^k_{\tau e} \tag{4.27}$$

$$\forall_{k \in K} : \sum_{e \in \mathtt{In}(s(\tau))} x^k_{\tau e} = 0, \sum_{e \in \mathtt{Out}(s(\tau))} x^k_{\tau e} = 1 \tag{4.28}$$

$$\forall_{k \in K} : \sum_{e \in \mathtt{Out}(t(\tau))} x^k_{\tau e} = 0, \sum_{e \in \mathtt{In}(t(\tau))} x^k_{\tau e} = 1 \tag{4.29}$$

$$\forall_{i \in N \setminus \{s(\tau), t(\tau)\}} \forall_{k \in K} : \sum_{e \in \mathtt{In}(i)} x^k_{\tau e} = \sum_{e \in \mathtt{Out}(i)} x^k_{\tau e} \tag{4.30}$$

$$\forall_{k \in K} : \sum_{e \in E} |e| \times x^k_{\tau e} \leq \lambda \tag{4.31}$$

$$\forall_{i \in N \setminus \{s(\tau), t(\tau)\}} : \sum_{e \in \mathtt{In}(i), k \in K} x^k_{\tau e} \leq 1 + (|K| - 1) \times z_{\tau i} \tag{4.32}$$

$$\forall_{e,a \in E | s(e) < t(e) \wedge s(e) = t(a) \wedge t(e) = s(a)} : \sum_{k \in K} x^k_{\tau e} + x^k_{\tau a} \leq 1 + (|K| - 1) \times q_{\tau e} \tag{4.33}$$

$$\forall_{i \in N} : \sum_{e \in E, k \in K | s(e) = i \wedge t(e) = i} x^k_{\tau e} \geq z_{\tau i} \tag{4.34}$$

$$\forall_{e \in E | s(e) < t(e)} : \sum_{k \in k} (x^k_{\tau e} + \sum_{a \in E | s(a) = t(e) \vee t(a) = s(e)} x^k_{\tau a}) \geq q_{\tau e} \tag{4.35}$$

$$\forall_{i \in N} : z_{\tau i} \in \{0, 1\} \tag{4.36}$$

$$\forall_{e \in E} : q_{\tau e} \in \{0, 1\} \tag{4.37}$$

$$\forall_{e \in E} \forall_{k \in K} : x^k_{\tau e} \in \{0, 1\} \tag{4.38}$$

Figure 4.3: FINDCHEAPESTBOUNDEDMAXIMALNODEDISJOINTPATHS(τ, N, L, L^b, K): FIND K CHEAPEST BOUNDED BETWEEN THE MC NODES OF WITH MAXIMAL DISJOINTNESSτ

	Network		
Criteria	UK	Italy	Spain
Metro core nodes	75	116	179
Number of pair of metro nodes	2775	6670	15931
Number of not node disjoint pairs in MDD abstract network	74	8	0
Percentage of Node Disjoint Pairs in the MDD abstract network	97%	99%	100%
MDD abstract network size, Step 2 (km)	11523.79	34639	25161.51
MDD street network Size, Step 2 (km)	8575.79	15228.19	16357.59
Number of links in the MDD street network, Step 2	138	269	326
Percentage of node disjoint pairs in the street network, Step 2	7%	11%	17%
Percentage of edge disjoint pairs in the street network, Step 2	14%	11%	18%
Ratio of total shared edge length (%) in the street network, Step 2	28%	37%	41%

Table 4.2: Size of the Minimum Distance Abstract and Street Networks after Step 2

4.1.3 STEP 3: Projection of Abstract Network Solution into the Street Network and Improving

Although it is most likely to find high node disjointness in the abstract level after STEP 2, as shown in Table 4.2 , due to not using actual trails of paths, some road junctions might be used by both disjoint paths. Therefore, the actual disjointness in the street level might be much smaller than the disjointness in the abstract level. However, since there are still many trails that can be used in the street level solution obtained in STEP 2 can be repaired. In this step, we provide an algorithm to repair paths of pair of metro-nodes which are labeled as not node-disjoint after projection in the reference network. Repairing is based changing the set of trails of an abstract link in the reference network so that the number of shared nodes between paths of a given pair of metro core nodes is minimised. Note that the repairing algorithm assumes that $K = 2$.

- Let P be the set of all pairs of MetroCore (MC) nodes.
- Let F^a_{ij} and S^a_{ij} denotes two node-disjoint paths between nodes i and j in the abstract path. Each path can be viewed as an ordered set of links.
- Let R^{fa}_{ij} be a set of pairs of MC-nodes such that the first path between each pair of MC-nodes relies on the link between i and j. Similarly R^{sa}_{ij} be a set of pairs of MC-nodes such that the second path between each pair of MC-nodes relies on the link between i and j.
- Let $G^r = \langle V^r, E^r \rangle$ be the graph corresponding to the street network.
- Let F^r_{ij} denotes the first path in the street network between MC-nodes i and j which is a set of ordered arcs. Let S^r_{ij} denotes the second path in the street network between MC-nodes i and j.
- Let $Q \subseteq M^2$ be the set of pairs of MC-nodes that do not have two node disjoint paths.
- Let $A(i, j)$ be an ordered set of edges associated with the path $i \in M$ and $j \in M$ in the street network.
- Let $N(P)$ be an ordered set of nodes associated with the path P in the street network where the source and the target nodes of the path are excluded.

- Let C_1^a and C_2^a refer to links in the F_{ij}^a and S_{ij}^a repectively.

The presented algorithm considers the abstract graph and repairs as many links as possible. The general idea is to first compute the set of pairs of metro-core nodes that are in conflict. For each such pair of metro-core node, first it tries to repair the first path and then the second path. For the first path the algorithm computes all the links that are source of violating disjoint constraint in the reference network. For each such link it iteratively tries to repair by taking in to account that other metro-core nodes might be relying on this link and the length of the path associated with any pair of node is within threshold. If an alternate road path is found then the mapping between abstract link and the road path changes and consequently the length of abstract links and the total cost. The same procedure is iterated for the links that are still in conflict in the second path of a given pair of nodes.

Result of the Step 3 is presented in Table 4.3. Under the assumption of using the same links found in Step 2, the percentage of node disjoint paths are increased from 7% to 66%, 11% to 40% and 17%to 58% for UK, Italy and Spain networks respectively. The cost of improving the node-disjointness ratio is increasing the size of the network 42%, 21% and 58% based on the reference network sizes in Step 2. Total length of the shared trails in the street network is also reduced almost 50% for each network.

	Network		
Criteria	UK	Italy	Spain
Metro core nodes	75	116	179
Number of pair of metro nodes	2775	6670	15931
Number of not node disjoint pairs in MDD abstract network	74	8	0
MDD abstract network size, Step 3 (km)	15814.31	37281.14	36185.96
MDD street network size, Step 3 (km)	12239.64	18442.67	25920.14
Number of links, Step 3	138	269	326
Diameter in the MDD abstract network, Step 3 (km)	2394.88	2429.98	2429.76
Percentage of node disjoint pairs in the street network, Step 3	66%	40%	58%
Percentage of edge disjoint pairs in the street network, Step 3	70%	41%	59%
Ratio of total shared edge length (%) in the street network, Step 3	12%	23%	19%

Table 4.3: Resilient designs for UK, Italy and Spain Networks

We also analyze the length of the working and protection paths in terms of utilising different optical signal types. Statistics in Table 4.4 show what percentage of the working/protection paths of the corresponding network can be served with each optical signal type. Results show that most of the working paths can be served with optical signals that have 1170km reach. However, the longest optical signals with 2430 km are the most suitable signals for the protection paths which are usually longer than the working paths. For example, for the working paths of UK network, 27.09% of the paths can be served with only the longest signals (i.e., 2430 km). Similarly, 36.14% of them are suitable for serving with the shortest signals with 500 km. However, 36.75% of them are too long to be served with 500 km signals and hence the optical signals with at least 1170km reach must be used.

while $Q \neq \emptyset$ **do**
  select and remove the pair of nodes $\{i, j\}$ from the set Q
  if F^r_{ij} and S^r_{ij} are not node-disjoint **then**
    $C^a_1 \leftarrow \{\langle p_1, q_1\rangle | \langle p_1, q_1\rangle \in F^a_{ij} \wedge \exists \langle p_2, q_2\rangle \in S^a_{ij}\ s.t.\ N(A(p_1, q_1)) \cap N(A(p_2, q_2)) \neq \emptyset\}$
    $B \leftarrow N(C^a_2)$
    for all $\langle p_1, q_1\rangle \in C^a_1$ **do**
      for all $\{l_1, l_2\} \in R^{fa}(p_1, q_1)\ s.t.\ N(A(p_1, q_1)) \cap N(A(p_2, q_2)) = \emptyset$ **do**
        $B \leftarrow B \cup N(S^r_{l_1 l_2})$
      for all $\{l_1, l_2\} \in R^{sa}(p_1, q_1)\ s.t.\ N(A(p_1, q_1)) \cap N(A(p_2, q_2)) = \emptyset$ **do**
        $B \leftarrow B \cup N(F^r_{l_1 l_2})$
      $temp(p_1, q_1) \leftarrow SP(p_1, q_1, G^r - B)$
      $feasible = true$
      for all $\{l_1, l_2\} \in R^{fa}(p_1, q_1)$ **do**
        for all $\langle p_1, q_1\rangle \in F^a_{l_1 l_2} \setminus C^a_1$ **do**
          $CurrentPath(l_1, l_2) \leftarrow CurrentPath(l_1, l_2) \cup A(p_1, q_1)$
        $CurrentPath(l_1, l_2) \leftarrow CurrentPath(l_1, l_2) \cup temp(p_1, q_1)$
        if $len(CurrentPath(l_1, l_2)) > \lambda$ **then**
          $feasible = false$
          $break$
      if $feasible$ **then**
        for all $\{l_1, l_2\} \in R^{sa}(p_1, q_1)$ **do**
          for all $\langle p_1, q_1\rangle \in S^a_{l_1 l_2} \setminus C^a_1$ **do**
            $CurrentPath(l_1, l_2) \leftarrow CurrentPath(l_1, l_2) \cup A(p_1, q_1)$
          $CurrentPath(l_1, l_2) \leftarrow CurrentPath(l_1, l_2) \cup temp(p_1, q_1)$
          if $len(CurrentPath(l_1, l_2)) > \lambda$ **then**
            $feasible = false$
            $break$
      if $feasible$ **then**
        $A(p_1, q_1) \leftarrow temp(p_1, q_1)$
        for all $\{l_1, l_2\} \in R^{fa}(p_1, q_1)$ **do**
          $A(l_1, l_2) = \emptyset$
          for all $\langle p_1, q_1\rangle \in F^a_{l_1 l_2} \setminus C^a_1$ **do**
            $A(l_1, l_2) \leftarrow A(l_1, l_2) \cup A(p_1, q_1)$
          $A(l_1, l_2) \leftarrow A(l_1, l_2) \cup A(p_1, q_1)$
        for all $\{l_1, l_2\} \in R^{sa}(p_1, q_1)$ **do**
          $A(l_1, l_2) = \emptyset$
          for all $\langle p_1, q_1\rangle \in S^a_{l_1 l_2} \setminus C^a_1$ **do**
            $A(l_1, l_2) \leftarrow A(l_1, l_2) \cup A(p_1, q_1)$
          $A(l_1, l_2) \leftarrow A(l_1, l_2) \cup A(p_1, q_1)$

	Percentage of Pairs can be Served with Different Signal Reaches					
	2430 km		1170 km		500 km	
Network	Working Path	Protection Path	Working Path	Protection Path	Working Path	Protection Path
UK	27.09%	39.89%	36.75%	41.90%	36.14%	18.19%
Italy	27.94%	70.37%	47.21%	18.60%	24.84%	11.01%
Spain	25.44%	55.10%	55.13%	27.50%	19.42%	17.38%

Table 4.4: Optical Signal Distribution for Working and Protection Paths

Finally, we would like to note that the following assumptions can be further relaxed to improve the percentage of the node disjoint pairs :

- A pair of MC-nodes can only be connected via one path in the street network or one link in the abstract network
- The abstract network is fixed that means the set of abstract links never changes while repairing
- The disjoint is not quantified. Therefore the path in the reference network associated with a link in the abstract network is modified only if it reduced the number of pairs of MC-nodes that have node-disjoint paths.

Furthermore, instead of finding the shortest bounded node disjoint paths in STEP 2, alternative bounded node paths can be generated and tested until a desired level of disjointness in the street network is achieved.

4.2 Resilient core network dimensioning using M/C nodes based on synthetic programmable ROADMs

In general, approaches for increasing network availability and reducing service disruption, along with the associated data and revenue losses, include *(i)* providing redundancy in the network to be used for failure recovery, and *(ii)* reducing the number of failure-prone components used by each optical connection (i.e., lightpath), thus lowering the associated risk of failure.

Recovery from link and node failures most commonly takes place at the *network level* by rerouting connections to disjoint paths when components at links or nodes included in the working paths of connections fail. Recently introduced synthetic reconfigurable add-drop multiplexers (ROADMs) implemented by Architecture on Demand (AoD) [21] offer new prospects of improving network reliability performance by supporting *node-level recovery*, i.e. self-healing from failures of node components. Recovery from optical node component failures at the node level, without triggering network-level recovery, favors the reuse of existing resources at the node rather than relying on extra spare resources in the network.

Technical requirements necessary to enable such functionality depend strongly on the optical node architecture. A common characteristic of various existing ROADM architectures [56] is that their constituent components are physically interconnected in a hard-wired manner. This gives rise to several drawbacks and limitations of such ROADMs, including restricted architectural flexibility, scalability and upgradability, as well as possibly inefficient

component usage which can inherently degrade lightpath availability and increase power consumption.

Recently, synthetic programmable optical switching nodes implemented by Architecture on Demand (AoD) have been proposed [20] in order to address the limitations of existing optical nodes by providing flexible processing and switching of optical signals through programmable architecture. In an AoD node, optical modules, such as optical splitters, amplifiers, or bandwidth-variable wavelength-selective switches (BV-WSSs) are interconnected through an optical backplane (i.e., optical switch) in contrast to a hard-wired manner as in conventional ROADMs. Deployment of AoD nodes has been shown to have superior performance compared to hard-wired ROADMs in terms of cost-efficiency [68], power consumption [51], scalability[50] and resiliency[40]. The concept of Architecture on Demand is suitable for the DISCUS architecture as the envisioned ROADMs deploy a high-port count Polatis switch which can be used as the optical backplane.

The prinicple of self-healing in AoD nodes is illustrated in Fig. 4.4. Inside the node, connections use only the components necessary for satisfying the processing requirements and bypass unneeded components, rendering them idle. When a component of a certain type fails, the optical backplane can easily be reconfigured to use an idle component of the same type located at the node, enabling fast failure recovery (down to 10 ms, i.e. the optical backplane switching time [96]) without activating network-wide survivability mechanisms. Note that the backup component can be an idle working component, or it can be added in the node specifically for failure recovery. In this way, AoD can provide redundancy for any type of components and supports addition of an arbitrary number of redundant components without disturbing the existing lightpaths.In the example shown in the figure, the traffic is supported by two splitters and two WSSs, while one splitter and one WSS remain in idle state. When working WSS 2 fails, the idle WSS (here WSS 3), takes over the failed connections.

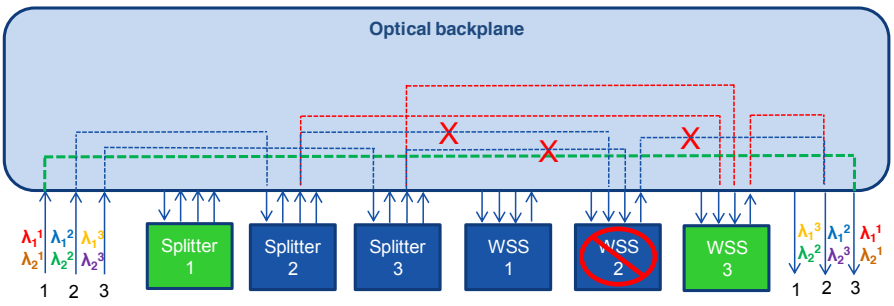


Figure 4.4: Self-healing in an AoD ROADM.

4.2.1 Enforced Fiber Switching (EFS) Routing Algorithm

A particularly beneficial AoD node functionality from the network reliability performance point of view is their ability to support switching at the fiber level, i.e., bypass all components between an input and an output port. This mechanism reduces the number of components traversed by each lightpath, which reduces the related risk of failure, and increases the number of idle components which can be reused for failure recovery [41]. With that in mind, we design a lightpath routing strategy called Enforced Fiber Switching (EFS) routing, which obtains a targeted portion of lightpaths undergoing fiber switching (FS) along their paths.

In order to utilize the adaptive nature of AoD nodes enabling minimization of the number of used components, we develop the Enforced Fibre Switching (EFS) routing algorithm. The goal of EFS is to improve lightpath availability by increasing the percentage of fibre-switched lightpaths at the network level. The subsequent increase in lightpath availability stems from two factors. To clarify, we consider the effects of establishing fibre switching between a single port pair in a node traversed by several lightpaths which use different pairs of input/output ports. Firstly, the lightpaths which are switched on the fibre level will bypass one active BV-WSS and one splitter in the considered node, which reduces their probability of failure and renders the bypassed components redundant. Secondly, lightpaths which are still undergoing switching at the wavelength level can now use one redundant BV-WSS and one redundant splitter for recovery in the case of failure of any working BV-WSS or splitter in the node. The impact of such redundancy and the changes it incurs in lightpath availability are visible from the availability models presented in Section VI-C. Adversely, some of the lightpaths which had to be rerouted to allow for the FS in the considered node might end up traversing a greater number of nodes than on their original, shortest path, comprising a greater number of total components, which might, in turn, decrease their availability. The strength of the EFS algorithm is precisely in balancing these two seemingly opposite effects, i.e., releasing sufficient redundancy in the network to compensate for the extra components traversed by the rerouted paths.

The pseudocode of the EFS algorithm is shown in Algorithm 2. In the beginning, EFS finds the shortest path in the physical topology for each lightpath and calculates the portion of lightpaths which undergo fibre switching at any node along their path (denoted as fs_temp). In highly-connected lightpath topologies, this value is typically very small (zero in most cases). Recall that, in order for FS to be possible between an incoming port from node *i* and an outgoing port to node *j* within node *n*, all lightpaths present at the port from *i* have to also be present at the port towards *j*. If this condition is not satisfied, lightpaths which are present on either port from *i* or port to *j* (extra lightpaths) have to be rerouted along an alternative path through the network in order to allow for FS between ports from node *i* to node *j*. The ratio between the number of lightpaths which are present on both port from *i* and port to *j* within node *n* (and can, therefore, be switched at the fibre level) and the number of lightpaths present only at port from *i* or port to *j* (and have to, therefore, be rerouted), is denoted as $fs\_ratio(n; i, j)$. In each step of the algorithm, we choose node *n* and its input-output node/port pair *(i,j)* with the highest value of $fs\_ratio(n; i, j)$ to perform FS. After re-routing of extra lightpaths, the procedure of finding the next node candidate is repeated until reaching the specified or the highest possible percentage of fibre-switched lightpaths.

In order to allow for fiber switching between input and output port pair (i,j) of node n, all lightpaths at port i must be connected to port j without passing any optical modules at the node. For densely interconnected lightpath topologies, this is not always possible if fixed shortest path (SP) routing is used. Consequently, in order to enforce FS, EFS must re-route some lightpaths to alternative, longer paths. When the probability of failures of physical links is taken into account, added fiber length and node components traversed by re-routed lightpaths might result in an undesired decrease of connection availability. Therefore, approaches for protection of AoD-based networks from failures of node components and links must pursue an advantageous trade-off between increasing the number of idle components which can be used as redundancy for failure recovery at the node level (thus increasing availability of certain connections while reducing the complexity of network-wide recovery management), and increasing the length of lightpaths re-routed to

Algorithm 2 Enforced fibre switching (EFS) routing algorithm.

Require: 1. Physical topology G = (E,V);
2. Lightpath demands τ = {s$_i$,d$_i$};
3. TFS = targeted portion of fibre-switched lightpaths;
Route lightpaths on their shortest paths and apply possible FS;
Calculate fs_temp = percentage of FS lightpaths in the network;
while $fs\_temp < TFS$ **do**
1. Find the node *n* and its input-output node pair (*i,j*) with the maximum $fs\_ratio(n;i,j)$;
if $fs\_ratio(n;i,j)$ = 0 for all $(n;i,j)$ **then**
break;
2. Apply FS between ports *i* and *j* and reroute extra lightpaths;
if applying FS or re-routing of extra lightpaths not possible **then**
break;
3. update fs_temp;

release this redundancy (leading to a decrease in availability). To this end, we develop a survivable routing algorithm for AoD-based networks called Dedicated Path Protection with Enforced Fiber Switching (DPP-EFS), which combines self-healing at the node level with 1+1 protection at the network level. To the best of our knowledge, no protection schemes have been developed so far for AoD-based networks.

4.2.2 Dedicated Path Protection with Enforced Fiber Switching (DPP-EFS)

Given a set of connection requests and a physical topology of an optical network based on AoD, the DPP-EFS algorithm must establish a pair of lightpaths, i.e., physically disjoint working and backup paths for each connection, while trying to maximize the portion of lightpaths undergoing FS along their paths. The algorithm begins by routing all working paths on the shortest paths, followed by an attempt to increase FS by rerouting certain paths to alternative routes. When no further increase in FS of the working paths is possible, the algorithm proceeds by establishing a shortest possible link-disjoint backup path for each connection request without violating the established FS. In order to reflect the fact that establishing FS restricts port connectivity, physical network G(V, E) is transformed into network G'(V', E') such that V' is the set of nodes corresponding to ports of nodes from V, and E' is the set of links modeling the connectivity of the set V'. If we denote nodes and edges from V and E as v_i and e_i, while v_i' and e_i' denote nodes and edges from V' and E', the steps for the network transformation can be summarized as follows:

1. Create a node v_i' corresponding to each input and output port of each node v_i".
2. Create edges e_i' between appropriate nodes v_i' and v_j' for each edge e_i. This maps the connectivity between nodes v_i in the original network topology.
3. To map connectivity inside individual nodes v_i, i.e. between nodes v_i' and v_j' corresponding respectively to an input and an output port of the same node v_i, do the following: if fiber switching is established between v_i' and v_j', connect v_i' only to v_j' by adding a new edge e_i'. If there is no FS between v_i' and any v_j', add edges e_i' to connect v_i' to all nodes v_j' which correspond to output ports of v_i that are not fiber-switched to any input port corresponding to a node v_k'.

An illustrative example of transforming a simple 5-node network is shown in Fig. 4.5. After routing the four lightpaths on the routes shown in the left part of the figure, FS is established in nodes B and D. The right part of the figure shows the network after transformation. Vertices 1 to 24 correspond to the input and the output ports of nodes A to E. The red lines denote FS between a pair of input/output ports inside the same network node. For example, vertices 5 and 9 of node B correspond to FS ports and are connected only to each other, because routing any connection from port 5 to any output port but 9 would violate the established FS. The same applies to ports 15 and 18 of node D. On the other hand, connectivity of ports not connected via FS to any other port is much greater, as can be seen, e.g., inside node E in the figure.

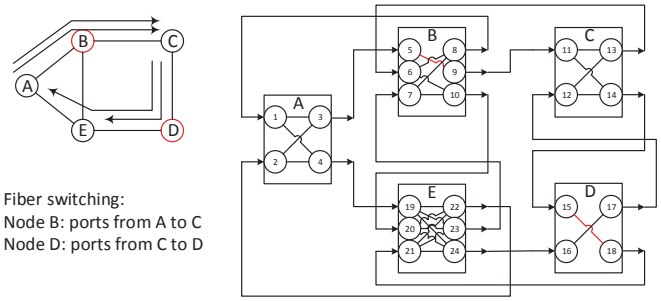


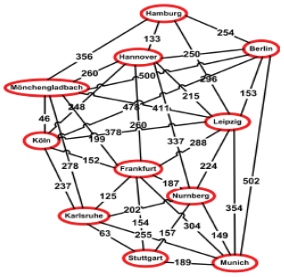
Figure 4.5: Transforming the network to reflect the impact of established FS to port connectivity.

After the network is transformed to incorporate the changes in connectivity introduced by FS, the backup path for each connection is found by deleting the nodes and links used by the working path and running Dijkstra's algorithm. We then evaluate the impact of the proposed approaches to network availability via simulation.

4.2.3 Numerical results

DPP-EFS was implemented in C++ and evaluated on a country-wide network of Germany with 11 nodes and 34 bidirectional links (Fig. 4.6 a)). Lightpath requests between node pairs were created to serve traffic which was generated proportionally to node populations and inversely proportional to their distances. The total traffic was scaled to model sparser and denser lightpath topologies. The algorithm was tested on 3 different test instances of each load. Failures of optical fibers, splitters, BV-WSSs and MEMS mirrors were generated using Monte Carlo simulations with exponentially distributed times to failure and to repair, under failure rates expressed in FIT (1 FIT = 1 failure in 10^9 hours) and summarized in Fig. 4.6 b), and a mean time to repair equal to 6 h. We simulated total time of 10^{10} hours and measured minimum (s,t) connection availability and mean down time (MDT) of the network, defined as the average number of minutes per year when at least one connection is in DOWN state due to failures. The results, shown in Fig. 4.7, were compared to a network deploying (i) hard-wired (H-W) and (ii) AoD ROADMs with DPP using SP routing for the working and the backup paths (denoted as DPP-SP). As shown in Fig. 4.7 a), the s,t availability in network deploying AoD under lower traffic load is greater than for H-W ROADMs even when SP routing is used, due to the fact that enough components remain idle and can be used for self-healing. When the logical topology becomes fully connected ($\geq$ 122 lightpaths), applying DPP-SP renders all components in AoD nodes utilized. In

such cases, combining AoD ROADMs with DPP-SP leads to slightly lower availability than H-W ROADMs due to the additionally traversed optical backplane elements. As can be seen from the total number of redundant, i.e., idle components in the network shown in Fig. 4.7 b), applying DPP-EFS releases extra redundancy in AoD nodes. To utilize this redundancy, DPP-EFS establishes on average 11% longer paths than DPP-SP. Consequently, DPP-EFS can heal more failures at the node level then DPP-SP applied to AoD ROADMs, which results in a lower number of failed connections, as shown in Fig. 4.7 c). Hence, implementing DPP-EFS in a network deploying AoD ROADMs results in higher s,t availability (Fig. 4.7 a) and 12.2% lower MDT (Fig. 4.7 d) than using DPP-SP in a network based on H-W ROADMs.



a)

Component	Failure rate in FIT
BV-WSS	10000
Splitter 1:N (N = degree)	25·N
MEMS mirror	21
Optical fiber	100/km

b)

Figure 4.6: (a) German backbone network and (b) component failure rates used in the simulations.

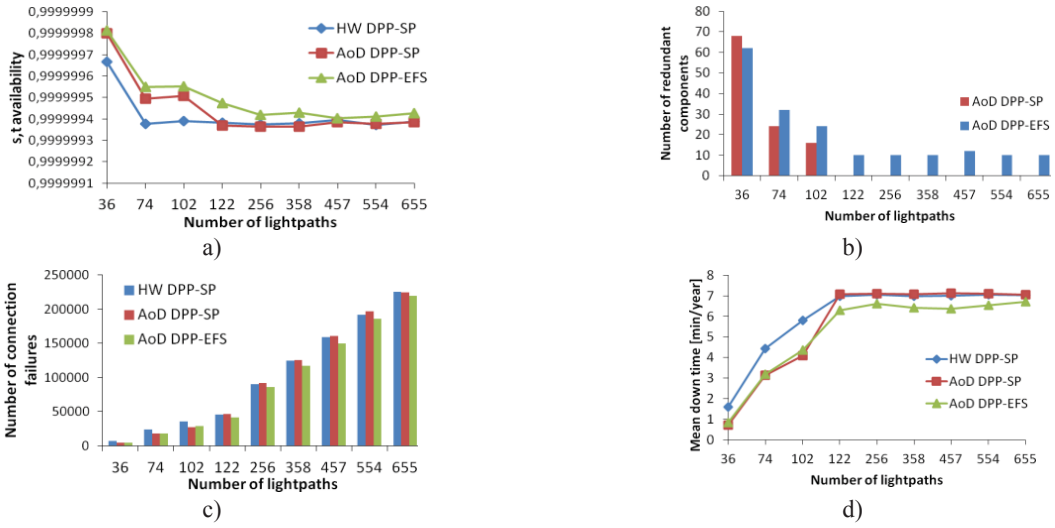


Figure 4.7: (a) s,t availability, (b) number of redundant components in the network, (c) number of failed connections and (d) network mean down time.

4.3 Protection of the core network in the presence of physical-layer attacks

Standard optical network protection approaches typically establish link and/or node-disjoint working and backup paths, focusing on fiber cuts and node equipment failures as the

leading causes of network interruptions. Although effective in the presence of failures, such approaches may not provide adequate protection in the presence of attacks due to their specific propagation characteristics. A number of optical-layer security breaches and physical-layer attack methods have been identified in the literature [58], [48], [84], [75], [49], [83], particularly in transparent optical networks. Attacks can induce financial losses to the clients or cause network-wide service disruption, possibly leading to huge data and revenue losses for operators. Attacks targeting service degradation, which are in the focus of this study, such as power jamming, typically involve inserting malicious signals into the network which can potentially propagate along configured connections causing widespread damage. To deal with this problem, in this section, we introduce the concept of an Attack Group (AG) defined as a set of connections which can affect each other in case they carry a malicious attacking signal. A connection is assumed to be attack-protected if its primary and backup paths are AG-disjoint. Protecting all connections from attacks could incur significant resource utilization and may not be economically viable. Thus, we incorporate attack-awareness into standard link-disjoint dedicated path protection to minimize the number of attack-unprotected connections only where little or no extra resources are required. We refer to the approach as Attack-Aware Dedicated Path Protection (AA-DPP). We develop a 2-step Integer Linear Program (ILP) formulation for AA-DPP, as well as an iterative heuristic for larger instances.

4.3.1 Propagation characteristics of service degradation attacks

Service degradation attacks generally involve inserting malicious signals into the network, such as optical signals of excessive power (e.g., 5-20 dB above other, legitimate signals), to degrade other user connections. Such a signal can be inserted on a legitimate channel used in the network (in-band jamming) or on a wavelength outside the signal window (out-of-band jamming) [83] and can degrade co-propagating user channels due to increased crosstalk and nonlinear effects in fibers. Furthermore, a jamming signal can cause so-called gain competition in optical amplifiers where the high-powered signal robs weaker legitimate signals of gain. Even if electronic equalization in amplifiers counteracts gain competition in the steady-state, initial brief oscillations in gain can occur. Additionally, in-band jamming signals can increase intra-channel crosstalk in optical switches to other user signals traversing the switch on the same wavelength. In a ROADM-based network envisioned by DISCUS, a jamming signal would be attenuated at intermediate nodes. However, some propagation may be achieved depending on the strength of the jamming signal and the working range of the associated VOAs. In Mixed Line Rate and elastic optical networks, an alternative service degradation attack could be achieved by inserting a lower line-rate OOK-modulated signal near a higher line-rate 40/100/200G channel using BPSK, QPSK or DP-QPSK modulation without allowing for enough guardband [49]. This could cause increased cross phase modulation effects significantly degrading the higher line-rate signals. If such a signal were inserted as a legitimate connection, it could propagate through the network degrading all co-propagating neighboring higher-rate channels. Such a signal would not be thwarted by power equalizing components or even be detected as malicious by power monitoring equipment.

4.3.2 Attack groups and the considered propagation model

In order to model the propagation characteristics of various attacks and incorporate them into the protection process, we introduce the concept of an Attack Group (AG). We de-

fine an Attack Group of a working or backup path i, denoted as AG(i), to be comprised of all other working paths which, if carrying an attacking signal, could potentially degrade path i. Naturally, this depends on the type of attack, its propagation characteristics and physical routing and wavelength assignment of all the connections in the network. The concept of AGs can be applied to individual attacks by considering their distinctive propagation characteristics or a combination of them. To cover for the worst-case scenario, we consider the propagation scenario where an attack inserted on any legitimate connection could potentially compromise all other connections sharing common links with it, as well as connections assigned the same wavelength and traversing common switches with it. Calculation of the Attack Groups of individual connections based on our propagation model and the impact of different path protection schemes is illustrated in Fig. 4.8. Assume four directed connection requests: c1:(5-4), c2:(5-6), c3:(4-2) and c4:(6-2); and three available wavelengths: λ_1, λ_2, and λ_3. Two different dedicated path protection schemes are shown in Figs. 4.8 a) and 4.8 b), where the working and backup paths of each connection c_i are denoted as c_i^W and c_i^B, respectively. In order to find the AG of each connection, we model attacking relations between connections with a so-called attack graph. In the attack graph, nodes represent the working and backup paths of individual connections and directed links indicate whether an attacking signal inserted on one of them can affect the other. Note, we assume 1:1 protection where only working paths can be the source of attack.

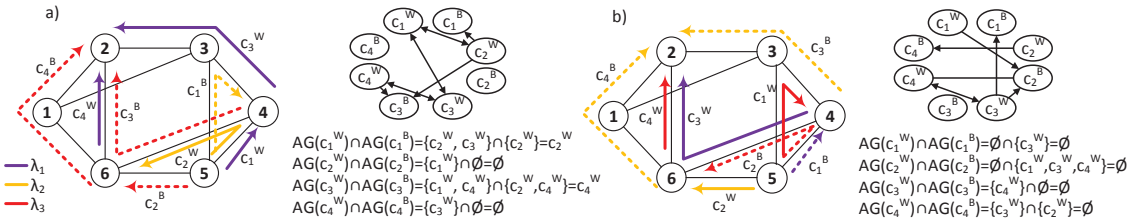


Figure 4.8: Two possible DPP schemes as their associated attack graphs.

The attack group of a working or backup path is then composed of all the nodes in the attack graph with which it is connected with via incoming links. For example, the attack group of working path c_1^W in Fig. 4.8 a) is $AG(c_1^W) = \{c_2^W, c_3^W\}$. Namely, connection c_1^W could potentially be degraded by an attack inserted on connection c_2^W since they share a common link (5-4), as well as by an attack inserted on c_3^W since they share a common switch (4) and are routed on the same wavelength (λ_1). To evaluate whether a connection is attack-protected, the attack groups of the primary and backup paths should be disjoint. The associated calculations of AG-disjointedness are shown in the figure. We can see that connection c_1 is not protected because its working and backup paths share a common AG element, i.e., c_2^W. Similar considerations apply to connection c_3, whose working and backup paths can both potentially be degraded by an attack inserted on c_4^W. Fig. 4.8 b) shows a slightly different path protection scheme which achieves better attack-protection while using the same number of wavelengths and total path length as the solution in Fig. 4.8 a). In Fig. 4.8 b), all connections are attack-protected, i.e. all connections have AG-disjoint working and backup paths.

4.3.3 Attack-Aware Dedicated Path Protection: Problem Definition

Given is a physical topology graph $G = (V, E)$, comprised of a set of nodes V interconnected by a set E of directional links. Given are also a set of available wavelengths Wand a set C of connection requests, each defined by their source and destination nodes. The Attack-Aware Dedicated Path Protection (AA-DPP) problem consists of finding a pair of link-disjoint paths for each connection request, along with their associated routing and wavelength assignment schemes, while minimizing the number of attack-unprotected connections. A connection is assumed attack-unprotected if the working and backup paths of the connection are not AG-disjoint according to the attack propagation model described above. Due to the high complexity of the AA-DPP problem with the described propagation model, we formulate a 2-step ILP which solves the routing and the wavelength assignment phases subsequently. The routing phase models attacking relations between connection sharing common links, while the wavelength assignment phase models attacking relations between connection traversing common switches on the same wavelength.

4.3.4 The 2-step ILP formulation (AA-DPP-ILP)

Due to the high complexity of the AA-DPP problem with the described propagation model, we formulate a 2-step ILP which solves the routing and the wavelength assignment phases subsequently. The routing phase models attacking relations between connection sharing common links, while the wavelength assignment phase models attacking relations between connection traversing common switches on the same wavelength. Note, an integrated ILP could be run by combining the two formulations to find a globally optimal solution but experimental results indicate that the integrated ILP could only be run in reasonable time for extremely small instances (4 nodes).

Step 1: The routing phase of AA-DPP-ILP

Notation and parameters:
$v \in V$: network nodes
$e \in E$: directed network links
$o_e, t_e \in V$: the source and destination node of link e, respectively
$c, d \in C$: connection requests
$o_c, t_c \in V$: the source and destination node of connection c, respectively
H: maximal total hop length

Routing variables:
$p_e^c = 1$ if the working path of connection c uses link e, 0 otherwise.
$\bar{p}_e^c = 1$ if the backup path of connection c uses link e, 0 otherwise.
$q_v^c = 1$ if the working path of connection c uses node v, 0 otherwise.
$\bar{q}_v^c = 1$ if the backup path of connection c uses node v, 0 otherwise.

Link-sharing and attack-reach variables:
$l_{c,d}^e = 1$ if the working path of connection c shares link e with the working path of connection d, 0 otherwise.
$\bar{l}_{c,d}^e = 1$ if the backup path of connection c shares link e with the working path of connection d, 0 otherwise.
$l_{c,d} = 1$ if the working path of connection c shares any link with the working path of con-

nection d, 0 otherwise.
$\bar{l}_{c,d} = 1$ if the backup path of connection c shares any link with the working path of connection d, 0 otherwise.
$a^L_{c,d} = 1$ if the working and the backup path of connection c both share link(s) with the working path of connection d (and can, thus, be attacked by d on links), 0 otherwise.
$a^L_c = 1$ if the backup path of connection c is attack-unprotected on links, 0 otherwise.

Objective: minimize the number of attack-unprotected connections under the assumption that an attack inserted on a working path can degrade any other working or backup path if they share a common link.

$$\min \sum_{c \in C} a^L_c \tag{4.39}$$

Subject to:

$$\forall_{c \in C}, \forall_{v \in V} : \sum_{e \in E : v = o_e, t_e} p^c_e = \begin{cases} 1 & \text{if } v = o_c \text{ or } v = t_c \\ 2q^c_v & \text{otherwise} \end{cases} \tag{4.40}$$

$$\forall_{c \in C}, \forall_{v \in V} : \sum_{e \in E : v = o_e, t_e} \bar{p}^c_e = \begin{cases} 1 & \text{if } v = o_c \text{ or } v = t_c \\ 2\bar{q}^c_v & \text{otherwise} \end{cases} \tag{4.41}$$

$$p^c_e + \bar{p}^c_e \leq 1, \forall_{c \in C}, \forall_{v \in V} \tag{4.42}$$

$$\sum_{c \in C} \sum_{e \in E} p^c_e + \bar{p}^c_e \leq H \tag{4.43}$$

Eqs. 4.40-4.41 define flow conservation and node usage, Eq. 4.42 ensures the working and back-up paths are link-disjoint and Eq. 4.43 limits the total hop length of all the paths to H.

$$l^e_{c,d} = p^c_e \wedge p^d_e, \forall_{e \in E}, \forall_{c,d \in C} \tag{4.44}$$

$$\bar{l}^e_{c,d} = \bar{p}^c_e \wedge p^d_e, \forall_{e \in E}, \forall_{c,d \in C}, \tag{4.45}$$

$$l_{c,d} = \bigvee_{e \in E} l^e_{c,d}, \forall_{c,d \in C} \tag{4.46}$$

$$\bar{l}_{c,d} = \bigvee_{e \in E} \bar{l}^e_{c,d}, \forall_{c,d \in C} \tag{4.47}$$

Eqs. 4.44-4.49 of AA-DPP-ILP ensure that if the working and the backup paths of a connection c both share links with the working path of any other connection, then connection c is marked as attack-unprotected. Eqs. 4.44 and 4.45 couple the working and the backup path of connection c, respectively, to any connection d whose working path shares a link with them, while 4.46 and 4.47 mark link-sharing over all connections. For the sake of brevity, symbols $\wedge$ and $\vee$ are used to denote logical AND and OR operations, respectively.

Relation $c = a \wedge b$ is implemented as $c \leq a$; $c \leq b$; $c \geq a + b - 1$. Similarly, relation $c = \vee_i a_i$ is implemented as $c \geq a_i, \forall_i$; $c \leq \sum_i a_i$.

$$a^L_{c,d} = l_{c,d} \wedge \bar{l}_{c,d}, \forall_{c,d \in C} \tag{4.48}$$

$$a^L_c = \bigvee_{d \in C} a^L_{c,d}, \forall_{c,d \in C} \tag{4.49}$$

Eq. 4.48 identifies connections c whose both the working and the backup path can be attacked by a connection d, while 4.49 denotes connection c as unprotected if any such connection d exists.

Step 2: The wavelength assignment phase of AA-DPP-ILP

In the Wavelength Assignment (WA) phase of AA-DPP-ILP, the values for the routing, link-sharing and attack-reach variables found by solving the routing phase of AA-DPP-ILP are used as input parameters, in addition to the set of nodes, links and connection requests. An additional parameter and the variables for the wavelength assignment phase are as follows

Notation and parameters:
$w \in W$: Set of available wavelengths

Wavelength assignment variables:
$r^c_w = 1$ if the working path of c uses wavelength w; 0 otherwise.
$\bar{r}^c_w = 1$ if the working path of c uses wavelength w; 0 otherwise.

Switch- and Wavelength-sharing and attack-reach variables:
$s^{c,d}_{v,w} = 1$ if the working path of connection c shares switch v and wavelength w with the working path of connection d, 0 otherwise.
$\bar{s}^{c,d}_{v,w} = 1$ if the backup path of connection c shares switch v and wavelength w with the working path of connection d, 0 otherwise.
$s^{c,d} = 1$ if the working path of connection c shares any common switch and a wavelength with the working path of connection d, 0 otherwise.
$\bar{s}^{c,d} = 1$ if the working path of connection c shares any common switch and a wavelength with the working path of connection d, 0 otherwise.
$a^{LS}_{c,d} = 1$ if the working path of d can attack the working path of c on links, and the backup path of c in switches, 0 otherwise.
$a^{SL}_{c,d} = 1$ if the working path of d can attack the working path of c in switches, and the backup path of c on links, 0 otherwise.
$a^S_{c,d} = 1$ if both the working and the backup path of connection c can be attacked by the working path of d in switches, 0 otherwise.
$a_{c,d} = 1$ if both the working and the backup path of connection c can be attacked by the working path of d, 0 otherwise.
$a_c = 1$ if connection c is attack-unprotected on links and/or switches, 0 otherwise.

Objective: minimize the number of connections which remain unprotectedfrom attacks (on links or inside switches) according to are assumed propagation model described in Section 4.3.1.

$$\min \sum_{c \in C} a_c \tag{4.50}$$

Subject to:

$$\sum_{w \in W} r_w^c = 1, \forall_{c \in C} \tag{4.51}$$

$$\sum_{w \in W} \bar{r}_w^c = 1, \forall_{c \in C} \tag{4.52}$$

$$\sum_{c \in C} p_e^c r_w^c + \sum_{c \in C} \bar{p}_e^c \bar{r}_w^c = 1, \forall_{e \in E}, \forall_{w \in W} \tag{4.53}$$

Equations 4.51-4.53 model the wavelength clash and continuity constraints.

$$s_{v,w}^{c,d} = q_v^c r_w^c \wedge q_v^d r_w^d, \forall_{c,d \in C}, \forall_{v \in V}, \forall_{w \in W} \tag{4.54}$$

$$\bar{s}_{v,w}^{c,d} = \bar{q}_v^c \bar{r}_w^c \wedge q_v^d r_w^d, \forall_{c,d \in C}, \forall_{v \in V}, \forall_{w \in W} \tag{4.55}$$

$$s^{c,d} = \bigvee_{v \in V, w \in W} s_{v,w}^{c,d}, \forall_{c,d \in C} \tag{4.56}$$

$$\bar{s}^{c,d} = \bigvee_{v \in V, w \in W} \bar{s}_{v,w}^{c,d}, \forall_{c,d \in C} \tag{4.57}$$

Eqs. 4.54-4.57 identify wavelength- and switch-sharing among working and backup path pairs which determine individual attack groups. Eqs. 4.54 and 4.55 couple the working and the backup path of connection c, respectively, to any connection d whose working path shares a wavelength and a switch with them, while 4.56 and 4.57 mark wavelength- and switch-sharing over all connections.

$$a_{c,d}^S = s^{c,d} \wedge \bar{s}^{c,d}, \forall_{c,d \in C} \tag{4.58}$$

$$a_{c,d}^{SL} = s^{c,d} \wedge \bar{l}_{c,d}, \forall_{c,d \in C} \tag{4.59}$$

$$a_{c,d}^{LS} = l_{c,d} \wedge \bar{s}^{c,d}, \forall_{c,d \in C} \tag{4.60}$$

$$a_{c,d} = a_{c,d}^S \vee a_{c,d}^L \vee a_{c,d}^{SL} \vee a_{c,d}^{LS}, \forall_{c,d \in C} \tag{4.61}$$

$$a_{c,d} = \bigvee_{d \in C} a_{c,d}, \forall_{c \in C} \tag{4.62}$$

Eqs. 4.58-4.62 identify all possibilities in which the working and the backup path of a connection c can both be affected by an attacking signal carried on the working path of

a connection d, i.e. check if the working and backup paths of a connection are attack group disjoint. Eq. 4.58 identifies connections c whose working and backup paths can both be attacked by such a connection d in switches. Eq. 4.59 identifies connections c whose working paths can be attacked by connection d in switches, while its backup path can be attacked by connection d in links (and vice versa for 4.60). Eq. 4.61 identifies connections c where both the working and backup path can be attacked by a connection d on links and/or in switches, while 4.62 denotes connection c as attack-unprotected if any such connection d exists.

4.3.5 Attack-Aware Dedicated Path Protection heuristic (AA-DPP-H)

For larger problem instances, solving the 2-step AA-DPP-ILP becomes computationally intractable. Therefore, we propose an iterative heuristic for attack-aware dedicated path protection, denoted as AA-DPP-H. The pseudocode of the AA-DPP-H algorithm is shown in Fig. 4.9. As in the ILP, the objective of AA-DPP-H is to minimize the number of attack-unprotected connections according to the propagation model described in Section 4.3.1. Since the heuristic offers more flexibility than the ILP, we apply a secondary objective to minimize the number of working paths that can simultaneously be affected by a single attack inserted on any one of them, referred to as the Attack Radius (AR). The AR corresponds to the maximal size of the attack group of any working path. For example, the AR of the example shown in Fig. 4.8 a) is $AR = max\{|AG(c_1^W)|, |AG(c_2^W)|, |AG(c_3^W)|, |AG(c_4^W)|\} = max\{2, 1, 2, 1\} = 2$. Analogously, the AR of the DPP scheme in Fig. 4.8 b) is equal to 1.

The AA-DPP-H algorithm takes as input the physical topology $G = (V, E)$, the number of available wavelengths W, the set of connection requests C, the maximum allowed number of iterations i_{MAX}, and the number of K shortest paths to be considered as candidate routes. In the beginning, the incumbent solution, denoted as C_{SOL}, is empty, while the number of unprotected connections (UC) and the AR are initialized to a large numerical value modeling infinity. The algorithm iteratively constructs a feasible solution C_{SOL_i} using a greedy attack-aware approach, and updates the incumbent solution if a more secure solution C_{SOL_i} is found. The algorithm ends if a solution where all connections are attack protected is found or the maximal number of iterations is reached. Details of the construction and evaluation phases are described below.

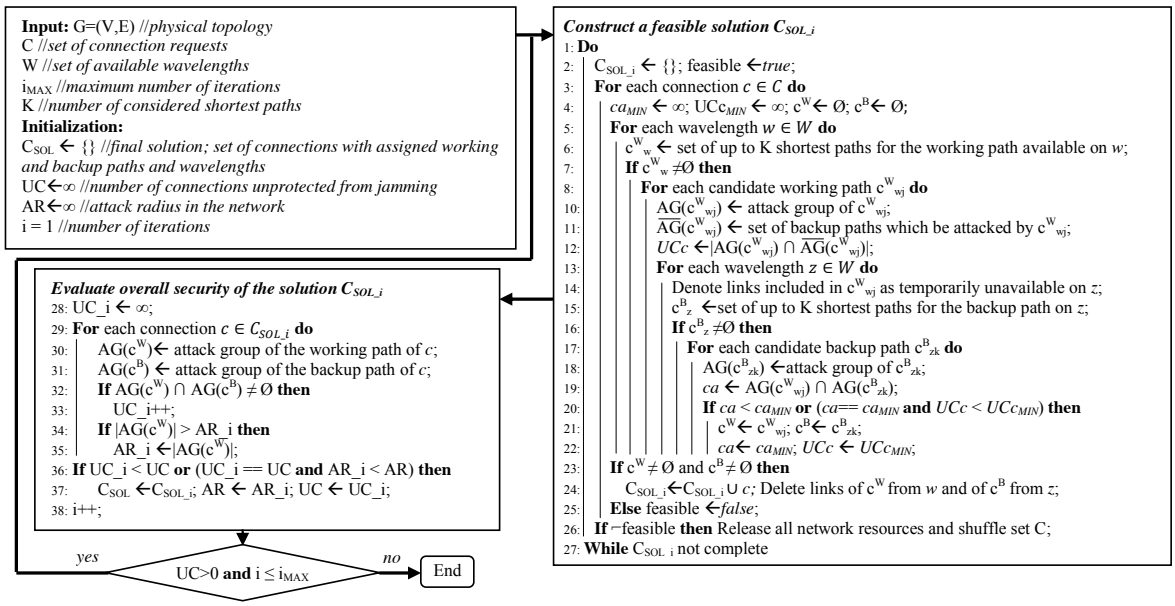


Figure 4.9: Flow-chart of the AA-DPP algorithm.

Construction phase

In order to construct a feasible solution, the algorithm processes one connection request c at a time while attempting to minimize *(i)* the number of potential common attackers shared among its working path c^W and backup path c^B (denoted as ca in the pseudocode); and *(ii)* the number of existing connections whose AG-disjointedness is violated by c^W (denoted as UCc). We use a layered graph approach where a copy of the physical topology is made for each wavelength, and occupied wavelength-links in the current partial solution are deleted from their corresponding layers when searching for paths for subsequent requests. For each connection request, the approach first searches for up to K available shortest path candidates for c^W on each available wavelength (row 6). For each c^W candidate, denoted as c^W_{wj}, the corresponding AG is identified (row 10), as well as the set of existing backup paths which can be attacked by c^W_{wj}, denoted as $\overline{AG}$ (row 11). The latter is needed to calculate the value of UCc as the number of elements in the intersection of sets AG and $\overline{AG}$ (row 12).

The algorithm proceeds by finding a set of up to K shortest path candidates for c^B on each of the available wavelengths, which are link-disjoint with c^W_{wj} (row 15). After calculating the AG of each c^B candidate, denoted as c^B_{zk} (row 18), common potential attackers on c^W_{wj} and c^B_{zk} are identified (row 19) and the pair of candidate paths with the minimum value of ca is selected as the solution for c^W and c^B (row 20). If the value of ca is the same for two candidate path pairs, the pair yielding lower UCc value is selected. The combined path length of c^W_{wj} and c^B_{zk} is used as a second tie-breaking rule to prioritize shorter paths (omitted from the pseudocode for the sake of brevity). Finally, if c^W and c^B are found, they are added to C_{SOL_i} and their links are marked as unavailable on the selected wavelengths w and z (row 24). If no feasible solution is found all resources are released and set C is shuffled randomly (row 26). The procedure is repeated until a feasible solution C_{SOL_i} is found.

Evaluation phase

In the evaluation phase, the feasible solution found in the construction phase is evaluated with respect to the previously described optimization criteria (i.e. number of attack-unprotected connection and the attack radius). The algorithm calculates the AGs for all working and backup paths (rows 30-31), identifies attack-unprotected connections (rows 32-33) and calculates the network AR (rows 34-35). If the current solution C_{SOL_i} has fewer attack-unprotected connections UC_i, or the same UC_i but a lower AR (row 36), the incumbent solution is updated (row 37).

4.3.6 Simulation results

To evaluate the efficiency of the proposed approaches, they are compared to standard attack-unaware DPP aimed at minimizing the lengths of the established paths and the number of wavelengths used. We consider 2 non-attack-aware benchmarking approaches, denoted as DPP-ILP and DPP-H.

DPP-ILP is a 2-step ILP formulation, analogous to AA-DPP-ILP, but instead of minimizing the number of attack-unprotected connections, the routing phase aims at minimizing the total path length used, while the wavelength assignment phase minimizes the number of wavelengths used. The formulation is derived from formulation AA-DPP-ILP as follows. The link/switch-sharing and attack-reach variables are dropped in both phases, as well as input parameters H and W. The routing sub-problem of DPP-ILP is then solved with the objective of minimizing the total length of the established paths, equal to the left-hand side of Eq. 4.43, and by considering only constraints 4.40-4.42 from 4.3.4. The WA sub-problem of DPP ILP is aimed at minimizing the number of used wavelengths. Thus, the formulation employs the wavelength assignment variables from 4.3.4 as well as an additional binary variable y_w indicating whether wavelength w is used in the final solution. The objective is then to minimize the total number of wavelengths used, i.e. $\sum_{w \in W} y_w$. The WA phase is solved by applying the constraints described in Eqs. 4.51-4.53, as well as the following 2 constraints which indicate the wavelengths used:

$$r_w^c \leq y_w, \forall_{c \in C}, \forall_{w \in W} \tag{4.63}$$

$$\bar{r}_w^c \leq y_w, \forall_{c \in C}, \forall_{w \in W} \tag{4.64}$$

DPP-H is a heuristic approach also run as an iterative process with different connection request orderings analogous to AA-DPP-H, but without considering attacks. In every iteration, it searches for a working and a backup path of all requests with the objective of minimizing the number of used wavelengths. The working and backup path of each connection are assigned the shortest path on the first available wavelength. In every iteration, the algorithm begins by using just one wavelength and adds a new one only when the working or the backup path cannot fit on any of the previously used wavelengths. At the end of each iteration, the incumbent solution is updated if a solution has been found using fewer wavelengths, or the same number of wavelengths as the incumbent, but shorter total path length.

The 2-step ILP formulations AA-DPP-ILP and DPP-ILP were solved using CPLEX v12.4, while the heuristics AA-DPP-H and DPP-H were implemented as a software tool in C++. All

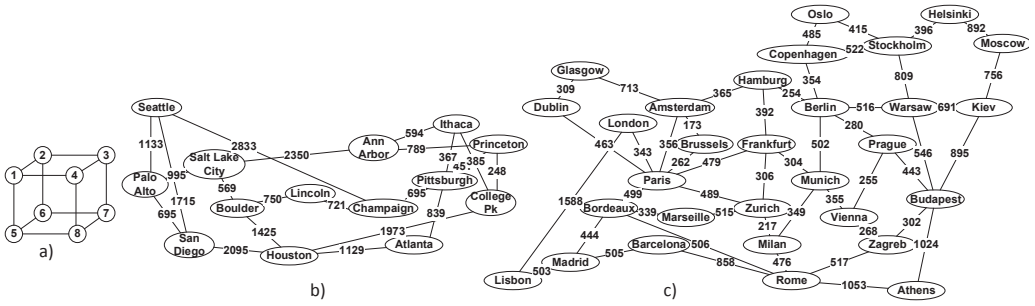


Figure 4.10: (a) The 8-node, (b) 14-node NSF, and (c) 28-node EON network used in the simulations.

algorithms were run on an HP workstation equipped with 8 Intel Xeon 2.67 GHz processors and 16 GB RAM. Three networks were considered, shown in Figure 4.10. Due to the complexity of the 2-step attack-aware ILP formulation, AA-DPP-ILP and DPP-ILP were only tested for the network shown in Fig. 4.10 a) comprising 8 nodes and 12 bi-directional links, each representing two directed links. The AA-DPP-H and DPP-H heuristics were evaluated on the 8-node network, as well as on the well-known reference networks shown in Figs. 4.10 b) and c) corresponding to the US NSF network (14 nodes, 21 bidirectional link) and European EON network (28 nodes, 41 bidirectional links), respectively. The traffic for the 8-node network was generated randomly, with the average logical degree equal to 2.86. Traffic for the NSF and EON network was generated based on the node population and distances, with the average number of lightpaths ranging from 125 to 250.

Fig. 4.11 shows the performance of all the implemented algorithms for the 8-node test cases. Fig. 4.11 a) shows the percentage of attack-unprotected connections for each test case. AA-DPP-ILP obtains solutions in which all connection are attack-protected, while AA-DPP-H leaves an average only 7% of connections attack-unprotected. In comparison, the attack-unaware approaches, DPP-ILP and DPP-H, obtain solutions with an average of 71% and 80% connections attack-unprotected, respectively. Figs. 4.11 b) and c) show resource consumption of the proposed approaches in terms of the total number of hops in the established paths and the number of used wavelengths. Since the routing phase of AA-DPP-ILP takes as input the total hop count obtained by the routing phase of DPP-ILP (whose objective is to minimize hop count), both formulations establish solutions with equal hops and are, thus, shown together in the figure. All the approaches gave similar results with respect to the total average hop count. The wavelength usage in the solutions obtained was also similar for all the approaches. Since the number of wavelengths used by DPP-ILP and DPP-H (whose objectives are to minimize wavelengths) were fed as input parameters to their attack-aware counterparts, they yielded the same number in all the cases tested and are, thus, shown together in the figure.

Fig. 4.12 shows the performance of the heuristics for the NSF and EON test cases. On average, DPP-H leaves more than 80% of connections unprotected from attacks in both networks, while AA-DPP-H reduces the number of vulnerable connections to only 3% in the NSF and 13% in the EON network. Since AA-DPP uses the same number of wavelengths as wavelength-minimizing DPP while establishing slightly shorter paths, we can conclude it represents a resource-efficient approach of reducing the potential damage from attacks.

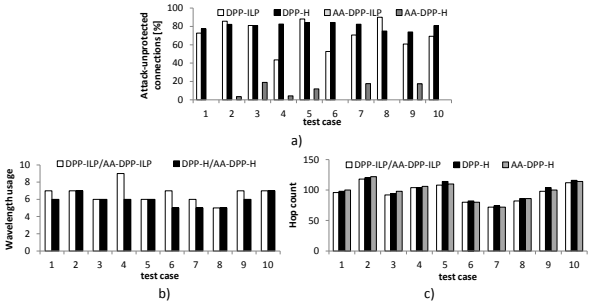


Figure 4.11: (a) Portion of attack-unprotected connections, (b) total path length and (c) number of wavelengths used by DPP-ILP, DPP-H, AA-DPP-ILP and AA-DPP-H in the 8-node network.

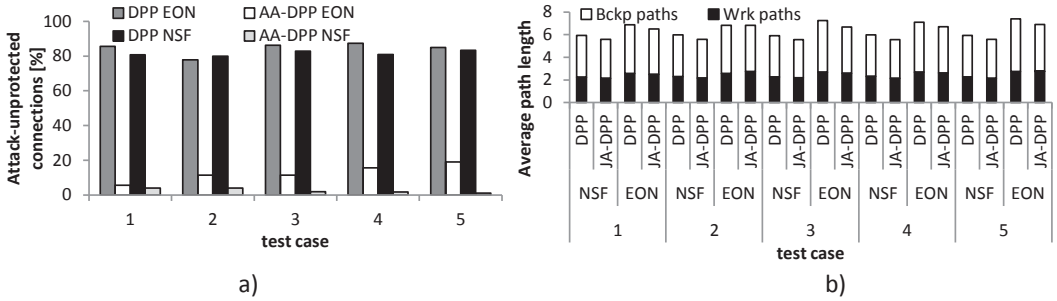


Figure 4.12: (a) Portion of attack-unprotected connections and (b) average path length obtained by DPP-H and AA-DPP-H in the NSF and EON network.

4.4 Resilience strategies based on dual homed M/C nodes

This section presents the benefits of dual-homing in the access to provide higher end-to-end resilience and better load-balancing, where each connection spans from one local exchange to another through the core network. As evident from Fig. 4.13, each LE connects to two MC nodes using dual-homing. While looking into the end-to-end availabilities it is also explored whether the path redundancies added by dual-homing play any role in providing efficient distribution of load across the core network and thereby reduce the cost of provisioning capacity in terms of number of lightpaths, transponders etc. The resilience approach mentioned here, focuses on the end-to-end philosophy where each communication request spans through the access link at the source side, followed by the core links, and then again an access link at the destination side. Dual homing at both the source and destination sides benefits survivability by providing more end-to-end path options. The results presented later, show dual-homed access proves to be advantageous over single-homed access by effectively utilising the end-to-end path alternatives.

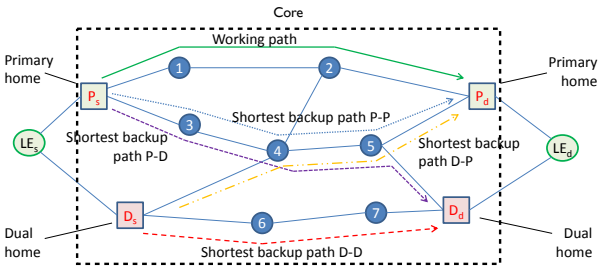


Figure 4.13: Example of different backup path options enabled by dual homing.

4.4.1 Mathematical model

The problem that we intend to solve, satisfies certain target availabilities of each end-to-end connection and also tries to minimise the network resources while providing a certain degree of resilience. We split the problem into two stages for sake of tractability. In the first step, we consider an extended network topology with both the access and the core links as illustrated in Fig. 4.13 and find a set of optimum end-to-end routes (i.e., one primary and one secondary path) that jointly meets a certain target availability. For that, we use a modified version of the basic link-flow model as described in [85]. We obtain a set of reliable routes for each connection and also the distribution of flows through the core network links from the first stage results. We utilise those results to obtain a minimum-cost (in terms of number of wavelengths and lightpaths) network provisioning in the second step. In the second stage our focus would be to design a transparent optical core network without any grooming or optical-to- electri-cal-to-optical (OEO) conversions. The transparency constraint can be relaxed to study variations of the current problem. The entire two-step problem is beyond the scope of the current deliverable. Currently a methodology to compute end-to-end routes that meet certain resiliency standards in a practical sized network is presented.

The mathematical formulation to find the optimum pair of end-to-end paths uses the following variables and notations.

$b_e^n = 1$ if link e connects node (local exchange) n to its primary home, 0 otherwise.
$\delta_{e,n} = 1$ if node n is an edge node of link e, 0 otherwise.
$x_{1,e}^{s,d} = 1$ if the working path $s-d$ uses link e, 0 otherwise.
$y_{1,n}^{s,d} = 1$ if working path of $s-d$ passes node n, 0 otherwise.
$w_{e1,e2}^{s,d} = 1$ if the working path $s-d$ uses link e_1, while the backup path $s-d$ uses link e_2, 0 otherwise.
$u^{s,d}$,u_e: The maximum allowed unavailability between source s and destination d and on link e respectively.
C: Total capacity in each link.
$T^{s,d}$: Traffic demand between local exchange s to local exchange d.
$x_{2,e}^{s,d} = 1$ if the backup path $s-d$ uses link e, 0 otherwise.

$y_{2,n}^{s,d} = 1$ if the backup path $s-d$ passes node n, 0 otherwise.

The MILP model of the problem is the following.

Objective:

$$Min \sum_{s,d} \sum_{e \in E} x_{1,e}^{s,d} + x_{2,e}^{s,d} \tag{4.65}$$

Subject to:

$$\sum_{e \in E: \delta_{e,n}=1} x_{1,e}^{s,d} = \begin{cases} 1 & \text{if } n = s \text{ or } n = d \\ 2y_{1,n}^{s,d} & \text{otherwise} \end{cases} \tag{4.66}$$

$$\sum_{e \in E: \delta_{e,n}=1} x_{2,e}^{s,d} = \begin{cases} 1 & \text{if } n = s \text{ or } n = d \\ 2y_{2,n}^{s,d} & \text{otherwise} \end{cases} \tag{4.67}$$

Constraints 4.66-4.67 are flow conservation constraints that maps the flow of links and the associated nodes.

$$x_{1,e}^{s,d} \leq 1 - b_e^s \quad \forall s, d \in V, e \in E \tag{4.68}$$

$$x_{1,e}^{s,d} \leq 1 - b_e^d \quad \forall s, d \in V, e \in E \tag{4.69}$$

Constraints 4.68-4.69 denotes that working paths must use the primary homes on both source and destination sides, i.e. are not allowed to use feeder fiber which connects it to the dual home.

$$x_{1,e}^{s,d} + x_{2,e}^{s,d} \leq 1 \quad \forall s, d \in V, e \in E \tag{4.70}$$

Constraint 4.70 accounts for link-disjointedness between primary and backup paths.

$$\sum_{s,d \in V} (x_{1,e}^{s,d} + x_{2,e}^{s,d}) \cdot T^{s,d} \leq C \quad \forall e \in E \tag{4.71}$$

$$\sum_{e_1 \in E} \sum_{e_2 \in E} w_{e_1,e_2}^{s,d} \cdot u_{e_1} \cdot u_{e_2} \leq u^{s,d} \quad \forall s, d \in V \tag{4.72}$$

Constraint 4.71 enforces the capacity constraint whereas constraint 4.72 is the end-to-end availability constraint. The MILP provides us with the flow of links that each end-to-end connection follows through the extended topology including both access and core.

The above ILP model is not scalable for practical network instances because the size of the problem is inherently large as we consider an end-to-end topology including the core network as well as thou- sands of local exchanges. Therefore, we propose a heuristic model described in Algorithm 3, to compute the availability-aware routes for each end-to-end connections.

The algorithm takes the extended topology including the MC nodes as well as local exchanges and their associated connectivities as in-put. It also takes an end-to-end demand

Algorithm 3 Heuristic for Link-Flow Model

Input: End-to-end topology $G = (V, E)$, traffic matrix $T = [\Lambda_{sd}]$, set of connections S, link availabilities set A, core network link capacity upper limit C, target availability A_t
Output: Set of two disjoint paths between each connection in S meeting target availability A_t.

1. Sort connections in descending orders based on their demand and put them in a list L.
2. for each connection in L, starting with the first connection, do (a) to (c) until no path found
   (a) Choose primary-home-to-primary-home path as working path.
   (b) Reserve capacities for the working path on the intermediate links and update residual capacities.
   (c) Sort 4 backup path options based on their ascending hop counts and store them in list BP.
   (d) for each of the 4 backup path options of the connection in BP, choose the first option and do:
      i. if availability of combination of working and backup path $\geq A_t$, do
         A. Check if residual capacities on the links of the chosen backup path can support the demand of the connection,
         B. If above condition is satisfied, assign the backup path and update the link capacities of each link in the backup path.
         C. Update residual capacities.
         D. if condition in (A) not satisfied, go to next backup option.
      ii. if availability of combination of working and backup path $\leq A_t$, go to next backup path option.
      iii. if availability and capacity condition is not satisfied, return 'no solution'.
3. calculate total path length for all working and backup paths.

matrix (T), the set of de-mands (S), the link availability set (A), the core network link capacity limit (C), a set of two disjoint shortest paths between each MC node pair, and the target availability (A_t) as input. The output of the algorithm returns a set of two disjoint paths between each connection in S meeting target availability A_t. The link flow model heuristic sorts the connections in descending order and for each connection it sets the primary-home-to-primary-home path for each connection as the working path (as shown in Fig. 4.13). Out of four choices of secondary or backup paths, the heuristic chooses the optimum path which meets both the link capacity requirement constrained by the capacity limit C and the target availability requirement i.e., the combined availability of the working and the backup path should be greater than the target availability A_t.

4.4.2 Numerical results

The results of dual-homing based resilience heuristic are presented in Table 4.5. The reference topology is a 20-node, 66-link. The different parameters for the illustrative results on the 6-node topology are as follows: (1) the sustained data rate has been assumed to be 10 Mbps per user; (2) the core network is assumed to have a capacity of 80 wavelengths per fiber with each wavelength carrying 40 Gbps of capacity; (3) The availability per kilometre of fiber is used to be 0.999979. End-to-end traffic is calculated using the well-known gravity model.

We observe that dual homing utilizes the core network resources in a better way than the single homing case. The number of hops and the number of links utilized in the dual homing case is less which is due to the fact that dual homing provides more options for end-to-end routing. Moreover, we also observe that the percentage of connections falling into higher availability class is more in case of dual homing than single homing for the same target availabilities. This proves that dual homing also provides better end-to-end resiliency.

The approach was also tested on a DISCUS UK reference topology and the results are shown in Table 4.6. The UK topology that we have used here is optimized based on minimum fiber distances and it is not always ensured that each MC-node pair has two disjoint paths. Therefore, some of the end-to-end routes only has one path if we do not consider dual homing. This degrades the availability considerably as we can notice from the table. Dual homing in this case proves to be significantly beneficial as it improves the average availabilities and hence the overall resilience.

4.5 Conclusion

In this chapter, we presented planning approaches for providing core-network survivability in the presence of single link- and node- failures, individual node component faults and deliberate physical-layer attacks targeting service disruption. We developed an abstraction-based design method for increasing the number of node pairs for which there are disjoint paths of minimized total length and satisfying the signal reach. Compared to the preliminary network design approaches which did not take resiliency into account, the new approach brings a significant increase in the number of node pairs connected by disjoint paths. To account for failures of individual components within nodes, we considered self-healing node architecture based on AoD and proposed a connection routing approach which alleviates the load of network-level recovery mechanisms by supporting

Scenarios	Total Link Usage	C1($A_t \geq 0.99999$)	C2 ($0.9999 \leq A_t < 0.99999$)	C3 ($0.999 \leq A_t < 0.9999$)	Average Primary Hop	Average Secondary Hop	Mean Availability
Dual Homing, $A_t = 0.99$	8901295	14.76%	85.22%	0.014%	6.39	6.05	0.9999784
Single Homing, $A_t = 0.99$	9438945	7.77%	92.23%	0	6.39	6.87	0.9999773
Dual Homing, $A_t = 0.999$	8601283	14.76%	85.22%	0.017%	6.39	6.05	0.9999785
Single Homing, $A_t = 0.999$	9438945	7.77%	92.23%	0	6.39	6.87	0.9999773
Dual Homing, $A_t = 0.9999$	8601270	14.76%	85.24%	0	6.39	6.05	0.9999786
Single Homing, $A_t = 0.9999$	9438945	7.77%	92.23%	0	6.39	6.87	0.9999773

Table 4.5: Dual Homing vs. Single Homing with the Irish topology

Scenarios	Total Link Usage	C1($A_t \geq 0.99$)	Average Primary Hop	Average Secondary Hop	Mean Availability
Dual Homing, $A_t = 0.95$	252746052	100%	6.83	6.44	0.9998
Single Homing, $A_t = 0.95$	254046744	24.38%	6.83	6.48	0.9848
Dual Homing, $A_t = 0.99$	252890573	100%	6.83	6.44	0.9998
Single Homing, $A_t = 0.99$	203578793	25.09%	6.83	6.44	0.9923

Table 4.6: Dual Homing vs. Single Homing with the UK topology

node-level recovery. The proposed approach brings a considerable portion of node component failures that can be healed at the node level and decreases the moan down time in the network. We also developed a protection approach to reduce the potential damage from deliberate physical-layer attacks targeting service disruption while maintaining resource-efficiency of standard, resource-minimizing protection approaches which consider only component faults. The developed approach dramatically reduces the number of connections which remain unprotected in the presence of attacks while requiring no extra resources. Finally, we presented a framework to calculate end-to-end resilient routes for a network architecture that consists both of a core network and an extended topology including access local exchanges. We compared two access architectures viz., dual homing vs. single homing and found that dual homing which has been an efficient resilience technique in the access can also provide better end-to-end resiliency and efficient load balancing in the core network.

Chapter 5

Resilient service provisioning

The resiliency work presented in Chapter 4 was mainly focused on design problems (i.e, the traffic matrix describing the services to be provisioned is known before hand). In this chapter we will consider, on the other hand, scenarios where services needs to be provisioned on the spot, without any prior knowledge of neither their arrival time nor their duration (i.e., a dynamic service provisioning scenarios).

Section 5.1 presents a study where the aim is to find hybrid network survivability strategies that combine the benefits of both protection- and restoration-based survivable schmes (i.e., high availability with low blocking rate). More specifically, the paper focuses on a double link failure scenario and proposes two strategies. The first one, couples dedicated path protection DPP (1:1) with path restoration (referred to as DPP+PR) to minimize the number of dropped connections. The second scheme adds up the concept of backup reprovisioning (BR), referred to as DPP+BR+PR, in order to further increase the connection availability achieved by DPP+PR.

Section 5.2 proposes a dynamic connection provisioning strategy which exploits the idle redundant modules use in survivable synthetic ROADMs to support failure recovery. The intuition behind the proposed approach is to use these modules to provision regular traffic with the aim of improving the blocking performance of the overall system.

Finally, Section 5.3 presents a heuristic based on the service relocation concept to be used for the dynamic restoration of optical cloud services. In other words, upon the occurrence of a network failure the proposed heuristic solves the routing and resource (i.e., transport plus cloud) allocation problem for each disrupted cloud service allowing, if necessary, to relocate some cloud services to different datacenter nodes.

5.1 Survivability Strategies WDM Networks Offering High Reliability Performance

Achieving high reliability performance is a key issue in survivable wavelength division multiplexing (WDM) networks. This is particularly important for mission-critical dynamic applications, where it is crucial to avoid data losses as a consequence of traffic-disruptions caused by failures. Path-protection-based survivability approaches providing fast recovery from fiber cut have been widely studied in the literature and dedicated path protection (DPP) is the most common strategy utilized by network operators, mainly because of its fast protection switching times [81] as well as its ease of design and implementation. However, path protection schemes are typically designed to provide 100% survivability against single link failures, while the occurrence of multiple link failures will degrade the network reliability performance. Obviously, adding more backup paths (e.g., DPP 1:N) would improve the reliability but it is costly, thus not preferred by network operators. In this context, backup reprovisioning (BR) approaches can be used to protect existing connections and to improve network reliability performance [89, 98]. The key idea is that after the first link

failure, BR is attempted for the vulnerable connections, i.e. the ones left without protection. If any of these connections will be affected by a second failure they will have a new backup path available. However, BR may not always be effective in multiple link failure scenarios, with total downtime values (i.e., the recovery time, if/when backup resources are available, or alternatively the remaining portion of the service time during which the connection is down, if a connection cannot recover from a failure) that tend to increase drastically.

This work presented in this section addresses this problem by involving path restoration (PR). We propose two failure recovery schemes, i.e., DPP+PR and DPP+BR+PR providing extra recovery options to vulnerable connections, thus maximizing the average connection availability. Time-efficient ILP models (i.e., with computation times less then 50ms) have been studied to implement optimal failure recovery for the proposed schemes in a PCE (Path Computation Element)-based [13] WDM network. Results show that, in a double link failure scenario, the proposed schemes can achieve higher connection availability, and lower downtime (compared to BR-based approaches), while showing far lower blocking probability compared to protection strategies based on multiple dedicated backup paths (i.e., DPP 1:2).

5.1.1 Survivable Provisioning Framework and Failure Recovery Schemes

This section first provides a general description of the survivable connection provisioning framework developed in this work. Then it describes the intuition behind the two proposed hybrid survivability schemes, i.e., DPP + PR and DPP + BR + PR. Finally, two additional schemes, i.e., DPP (1:2) and DPP + BR, used for benchmarking purposes are also introduced and described. For space reason the details about the ILP formulation of the various schemes are omitted but can be found in [17].

Framework for Survivable Connection Provisioning

The work in this section assumes a dynamic traffic provisioning scenario, where no more than two failures, regardless of their type, can affect the network simultaneously, i.e., only a double failure scenario in the network is considered.

Fig. 5.1 illustrates the finite state machine describing the various states in which a connection can be when provisioned in the network. Transitions among states are possible in the presence of specific events: (i) a failure happening in the network, (ii) a failed element being successfully repaired, (iii) a backup reprovisioning attempt being successful or unsuccessful, and (iv) a path restoration attempt being successfully/unsuccessfully made for a specific connection. Upon the occurrence of a failure, a connection can find itself in a number of different states depending on what is its current status, the failure recovery scheme adopted, and on the occurrence of other events (i.e., reparation and/or additional failure) in the network. More details are provided next.

According to the proposed framework, a connection is established along with a reserved, dedicated protection path (i.e., DPP). This guarantees the ability to survive to at least one failure, i.e., the connection is in a PROTECTED state. If a failure affects a protected connection, then a transition to the VULNERABLE state takes place. This state is used to characterize a connection that is still working normally but it is vulnerable against a possible additional failure striking the network, i.e., while already another failure is under reparation. Note that a connection ends up in the VULNERABLE state regardless of which

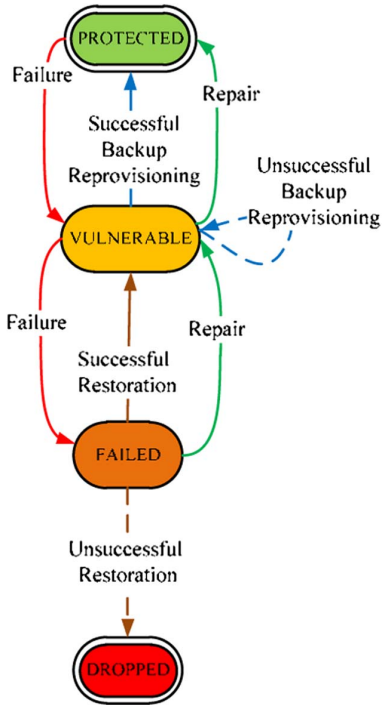


Figure 5.1: Finite State Machine showing: the possible states of a connection and the type of transition a connection can experience while provisioned in the network.

path (i.e., primary or backup) is affected by the failure. One possibility to get back to the PROTECTED state is via a backup reprovisioning attempt. If the attempt is not successful (i.e., lack of wavelength resources) then the connection stays in the VULNERABLE state until the failure is repaired. In the meantime, if another failure affects this vulnerable connection, then a transition to the FAILED state takes place. This state is used to represent a connection that is interrupted, i.e., the services provisioned via this connection are down. At this point two options are available to get back to the VULNERABLE state. The first one is to do a path restoration attempt. If path restoration is not successful (i.e., lack of wavelength resources) then the connection ends up in the DROPPED state. Once a connection is dropped, it cannot be recovered and its wavelength resources are released. Restoration is chosen in this framework for its reactive nature and for its inherent ability to efficiently use wavelength resources, i.e., backup paths are computed only after a failure occurs, and no backup resources are reserved beforehand. The other option to go back to the VULNERABLE state is to just wait until one of the two failures is repaired. This option might not be always viable, especially for services with strict downtime requirements.

Using the framework described so far it is possible to propose a number of hybrid survivability schemes that combine one or more of the transitions options just described. The choice depends on which performance measure (i.e., number of dropped connections vs. average connection downtime vs. efficient use of network resources) or combination of them is to be addressed. In this section we propose specifically two survivability schemes, i.e. DPP + PR and DPP + BR + PR. Finally, the last two survivability schemes described in this section, i.e., DPP (1:2) and DPP + BR, will be used for benchmarking purposes.

Dedicated Path Protection + Path Restoration (DPP + PR). The DPP + PR scheme tries to minimize the number of dropped connections while limiting the amount of occupied wavelength resources overall in the network. This is achieved by attempting path

restoration each time a connection is in the FAILED state but no backup reprovisioning is attempted in any case. On the other hand, as it was already pointed out, path restoration does't give any guarantee to be successful, especially at a medium or high network load where wavelength resources might become scarce. In addition, the recovery time of path restoration is relatively long. Both these aspects are addressed in the scheme described next.

Dedicated Path Protection + Backup Reprovisioning + Path Restoration (DPP + BR + PR). The DPP + BR + PR scheme uses the same principle, and has the same objective as the DPP + PR scheme, i.e., path restoration is attempted for each connection in the FAILED state. This is done in order to limit the number of dropped connections. However, the DPP + BR + PR scheme has an additional feature. In order to limit the number of connections that ultimately will require path restoration, each time a connection ends up in the VULNERABLE state backup reprovisioning is attempted. This is done with the intent of reducing the number of connections that, upon the occurrence of a second failure, will transition to the FAILED state and need path restoration. We expect that it would improve the average recovery time since protection-based mechanisms are typically faster than restoration-based techniques. As a result DPP + BR + PR has the ability to provide lower downtime compared to DPP + PR. On the other hand, backup reprovisioning comes at the cost of additional backup resources being reserved in the network.

Dedicated Path Protection (1:2). The DPP (1:2) [80] is our first benchmarking scheme. It assigns two mutually disjoint protection paths to each primary path. In this way DPP (1:2) guarantees to have connections always in the PROTECTED state as long as a single or double failure scenario is considered, i.e., there is always a protection path available. DPP (1:2) is also rather fast because switches on the intermediate nodes can be preconfigured for the backup paths ahead of time [81]. All these benefits come, on the other hand, at the expense of very high wavelength resources usage.

Dedicated Path Protection + Backup Reprovisioning (DPP + BR). The DPP + BR [89, 98] is our second benchmarking scheme. With DPP + BR, each time a connection becomes vulnerable backup reprovisioning is attempted once, in order to bring the connection back to the PROTECTED state. If backup reprovisioning fails the connection stays in the VULNERABLE state with the risk that, if affected by another failure, the connection becomes DROPPED, i.e., transition back from the FAILED state to VULNERABLE state may not be possible for this survivability scheme. It is important to note that upon the occurrence of a failure not just one, but a considerable number of connections can be potentially disrupted simultaneously. It will be then up to the selected survivability scheme to find an alternate backup route for each affected connection. In such a scenario concurrent optimization schemes have already been proven to be very efficient in finding routing solutions that optimize the resource usage in the network [16].

5.1.2 Simulation Setup and Numerical Results

Results are obtained using a Java-based discrete event-driven simulator running on a Red Hat Enterprise Linux workstation with 12 GB of memory and considering the NSF network topology [32], modified to become 3-edge-connected. All fiber links are bidirectional, with 16 wavelengths per fiber. Each lightpath is assumed to require an entire wavelength bandwidth. The presented results are the average of 32 replications. The connection holding time is exponentially distributed with an average equal to 1 time-unit. Moreover, Poisson arrivals of connection requests are considered assuming a uniform load per node pair. The

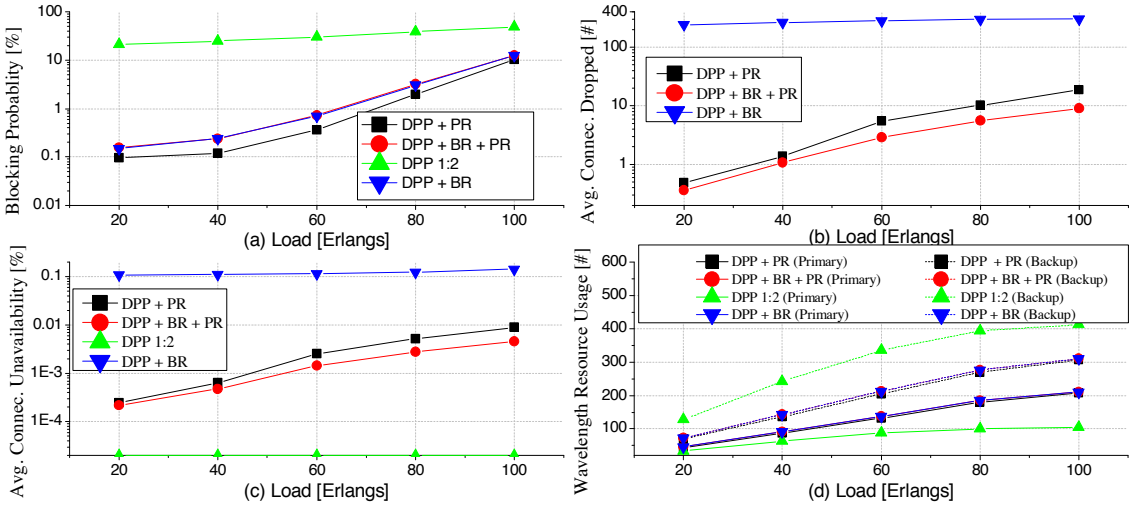


Figure 5.2: Blocking Probability (a), Avg. # of Dropped Connections (b), Avg. Connection Unavailability (c), Number of Used Wavelengths (d) as a function of the network load.

ILP models are solved using the Gurobi Optimizer 4.51 [1]. Time between failures occurring in the whole network is assumed to be exponentially distributed with an average equal to 2.5 time-units. Mean time to repair (MTTR) of a broken link is considered to be equal to 0.5 time-units. For dynamic DPP 1:1 connection provisioning, the heuristic presented in [18] is used. During failure recovery, the original primary path is restored back after a failed link is repaired. α, β, $\gamma 1$ and $\gamma 2$ in the ILP objective functions presented in [17] are assumed to be 10,000, 1.0, 0.5 and 0.25 respectively.

Fig. 5.2(a) shows that the blocking probability (BP) is substantially higher for DPP 1:2 because it provisions three mutually link-disjoint paths (one primary and two backups) per each connection. It is also shown that DPP+BR+PR has slightly worse BP performance compared to DPP+PR because of the BR operations. DPP+PR drops almost twice as many connections as DPP+BR+PR (Fig. 5.2(b)), but still, both DPP+PR and DPP+BR+PR have a substantially fewer number of connections dropped than DPP+BR. Both proposed schemes show low connection unavailability values (Fig. 5.2(c)) which are worse than DPP 1:2, but way better than DPP+BR. Finally (Fig. 5.2(d)), DPP 1:2 requires significantly more backup resources as compared to the two proposed schemes. The table reported in Fig. 5.3 shows the connection downtime values experienced when using different failure recovery schemes. The connection downtime is defined as the time in which a connection is not in normal working conditions because of a protection switching event, a restoration attempt being under way, or because of the connection being dropped. As can be expected from the connection unavailability results presented in Fig. 5.2(c), the DPP+BR scheme shows the worst downtime values because of the lack of any dynamic restoration procedure to recover from a failure after an unsuccessful backup reprovisioning attempt. On the other hand, the DPP + PR and the DPP + BR + PR schemes show much lower downtime values. Note that the downtime results for DPP (1:2) are not shown because of their negligible values since the only time a connection is down is during protection switching.

	DPP+PR			DPP+BR+PR			DPP+BR		
Load [Erlangs]	20	60	100	20	60	100	20	60	100
Downtime [time-units]	0.415	5.31	17.56	0.345	3.34	8.68	273	284	309

Figure 5.3: Total connection downtime for different schemes.

5.2 Dynamic Provisioning Utilizing Redundant Modules in Elastic Optical Networks Based on Architecture on Demand Nodes

In order to efficiently accommodate bandwidth-intensive and dynamic network applications, as well as legacy low bitrate demands, the network infrastructure, and in particular optical nodes, need to be flexible, cost-effective and reliable. Among different proposed solutions for elastic nodes, the ones based on Architecture on Demand (AoD) [46] exhibits remarkable flexibility compared to the existing alternatives [47].

AoD nodes support the switching of optical signals from an input port directly to their targeted output port through the optical backplane without utilizing any (de)multiplexing modules, i.e., an operation called fiber switching. This aspect of AoD has been exploited for the cost-efficient design of nodes [45] and networks [67]. Enhancing fiber switching improves network availability, as a decrease in the number of components traversed by a connection reduces the related risk of failure. AoD nodes can also replace failed components with idle (spare) modules on-the-fly for self-healing purposes.

On the other hand, applying cost-efficient network planning techniques [67] based on replacing modules with fiber switching might lead to a degradation of the network blocking performance in dynamic network conditions, where connection requests arrive and depart stochastically. In general, connection blocking in dynamic networks based on AoD nodes can be caused by two factors: (i) an insufficient number of switching modules at AoD nodes which cannot support the required connectivity; and (ii) a shortage of spectrum resources along the route between the source and destination node of a connection. Blocking inflicted by the former factor can be prevented by increasing the number of switching modules in an AoD node. Due to the flexibility of AoD nodes, another option is to utilize modules dedicated for protection to serve connection requests that would otherwise be blocked due to the restricted port connectivity. However, such utilization of redundant components might render them unavailable for accepting traffic from failed working components within the node, which can in turn decrease connection availability.

Inspired by these observations, this section presents a dynamic connection provisioning strategy aimed at improving blocking probability caused by the scarcity of switching modules in AoD nodes, while balancing the trade-off with connection availability. The strategy utilizes redundant switching modules deployed in a survivable AoD nodes to accommodate the requests which would otherwise been blocked. By preempting the connections established by a redundant module in order to protect connections served by a failed working module, the proposed strategy is capable of obtaining an advantageous trade-off between blocking probability and network availability.

5.2.1 Using Preemptable Modules in an AoD Node Architecture

The concept behind the proposed connection provisioning strategy is illustrated in Fig. 5.4.a, showing a possible configuration of an AoD node with nodal degree 4 support-

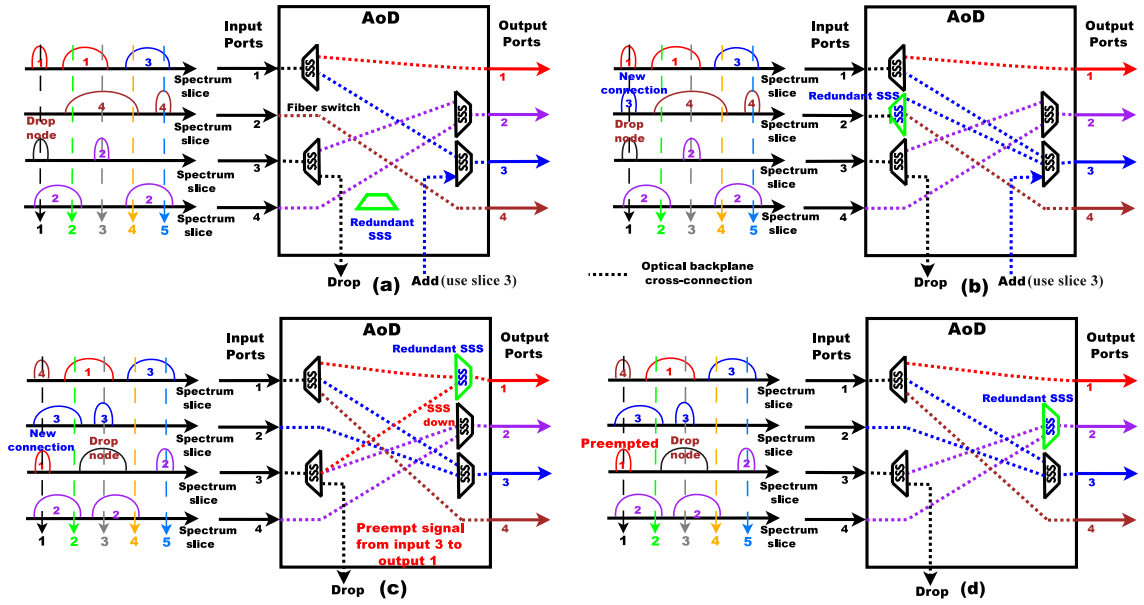


Figure 5.4: Sample scenarios for AoD node with a redundant module.

ing a set of established connections, assuming that each input fiber link has five spectrum slices. The three connections at input port 1 are fed to a spectrum selective switch (SSS) to allow flexible spectrum switching towards their respective output ports (i.e., one connection using slices 4-5 switched to port 3, and two connections using slice 1, and slices 2-3, respectively, are switched to port 1). The two connections at input port 2 are both directed to output port 4 and can be forwarded via a single cross-connect operation (i.e., fiber switching). Connections at input port 3 are fed to a SSS where the one using slice 1 is locally dropped, and the one using slice 3 is sent to output port 2. The connections at input port 4 are cross-connected directly to output port 2.

Aside from the four working SSSs needed to support the existing traffic, the node is also equipped with one redundant SSS to be used in case a working SSS fails. Suppose now that a new connection on slice 1 appears at input port 2, with output port 3 as its destination (Fig. 5.4.b). Assuming a node configuration as in Fig. 5.4.a, this new connection would have been blocked due to the absence of the required SSS at input port 2. However, the redundant, currently idle SSS can be put into use at input port 2 to accommodate this new connection (Fig. 5.4.b). If no failure happens while the spare SSS is used at input port 2, it will stay in operation as long as needed to switch the incoming traffic at input port 2 (e.g., until the connections directed to output port 4 depart from the network), after which it will go back to idle state.

Similarly, at some other time instance the redundant SSS might be used at output port 1 (Fig. 5.4.c) in order to multiplex a new connection from input port 3 which would otherwise have been blocked. If the working SSS at output port 2 fails while the redundant SSS is used, the redundant SSS will need to be disconnected from output port 1 and then used at output port 2. As a result connection using slice 1 at input port 3 is dropped (Fig. 5.4.d). Upon successful reparation of the failed SSS, the redundant module is detached from output port 2 and is put back to idle state, ready to be used again when needed. This example highlights how a redundant module in an AoD node can be deployed to reduce the number of blocked connections at the expense of a possible degradation of connection availability. A provisioning strategy aimed at achieving a beneficial trade-off between these

two objectives is presented next.

5.2.2 Dynamic Provisioning with Preemptable Spectrum Selective Switches (DP-PSSS)

This section describes how the concept of preemptable modules introduced in the previous section can be integrated in a dynamic provisioning strategy for AoD-based optical networks.

The connection set-up strategy presented in this section (i.e., Dynamic Provisioning with Preemptable Spectrum Selective Switches, or DP-PSSS) works as follows. Given the source and the destination node of a connection to be set up, DP-PSSS checks, for each route within a set of pre-computed k-shortest paths candidate, if all the AoD nodes traversed by a candidate path are capable to switch the signal from the required input to the required output port. Note that once fiber switching is used to connect an input port i with an output port j at a given node, it is no longer possible to switch signals from port i to any output port different from j as long as the connection(s) that is (are) fiber-switched between ports i and j are active. If the required connectivity at a given node along a candidate route is not supported due to fiber switching, DP-PSSS checks if there is an idle redundant SSS in that node. If yes, then the redundant SSS is used to switch the connection. If no spare SSSs are available, the candidate path is removed from further consideration. If there are no viable candidate paths for the current request, the connection is blocked. Otherwise, the spectrum availability is checked on each of the viable candidate paths. If free, continuous, and contiguous spectrum slices are found, the path remains a viable candidate route for provisioning the connection request. Among all the viable candidate routes, the one that requires a smaller number of new, previously unused SSSs is selected.

5.2.3 Numerical Results

A custom-built event-driven simulator is used to study the performance of the DP-PSSS strategy. Simulations are carried out on the NSF topology [42], with 14 nodes and 42 unidirectional fiber links, each supporting 80 spectrum slices. It is assumed that only single SSS failure can occur in the network, i.e., the probability that two or more SSSs will fail at the same time is considered to be negligible. Connections are assumed to arrive in the network following a Poisson process, each one requiring a number of slices uniformly distributed among $\{2, 4, 6, 8\}$ with an exponentially distributed duration, whose average value is set to 1 time unit. Source/destination pairs of connection requests are assumed to be uniformly distributed among all nodes. The number of candidate paths for each connection request is set to 5. The performance of DP-PSSS is assessed in terms of the number of demands rejected due to a limited number of switching modules at AoD nodes, and in terms of average connection availability. Results are compared against a benchmark strategy that works exactly as DP-PSSS, but does not allow using redundant SSSs to provision regular traffic. The presented results are averaged over fifty experiments for each traffic load value.

The number of active SSSs initially placed at each node is determined by the offline design procedure described in [42], where the network is dimensioned differently for each specific traffic condition. This translates in an overall number of switching modules in the network that is relatively small for low loads, and increases for higher load values. Such approach avoids over-provisioning of switching modules in nodes when and where they

are not needed. Finally, redundant SSSs are assumed to be placed only in those AoD nodes where at least one SSS is placed as a result of the design phase.

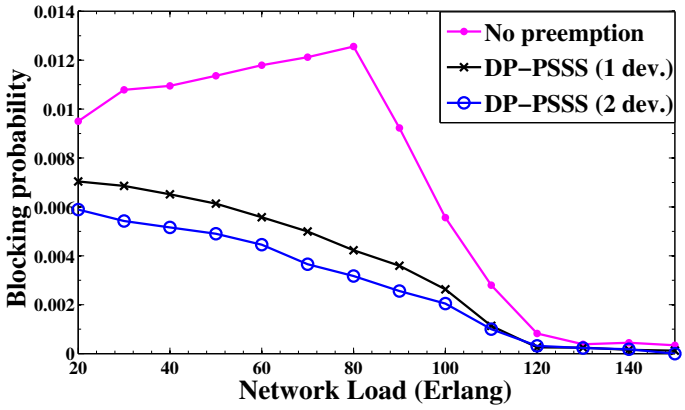


Figure 5.5: Blocking probability vs. offered network load.

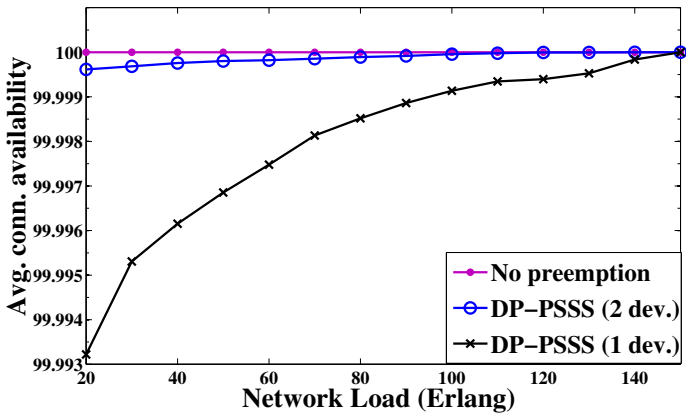


Figure 5.6: Average connection availability vs. offered network load.

Fig. 5.5 shows the average network blocking probability for two scenarios, i.e., when AoD nodes can have at most one or two redundant SSSs. For each load value the dynamic provisioning of connections is simulated with the number and placement of SSSs determined in the design phase. The blocking probability of the benchmark strategy first slowly rises and then sharply declines beyond 80 Erlang. This behavior is due to the fact that at low loads the number of active SSSs placed by the design phase described above is relatively small [67], yielding more restricted intra-nodal connectivity, while both parameters increase for higher load values. DP-PSSS achieves a significant reduction in blocking probability, i.e., an average gain of 53% and 62% with one and two redundant modules, respectively. The amount of spare resources depends on the load at which the network is designed. For the case of DP-PSSS with one redundant SSS, the number of spare SSSs reaches 40% of the total deployed working modules at low loads. At medium loads, this number drops to 25%. On the other hand, the number of spare SSSs reaches 70% at low and 50% of total working modules at medium load conditions, respectively, for the case of DP-PSSS with two redundant SSSs.

Fig. 5.6 shows the average connection availability (i.e., the ratio between the time a connection is operative over the entire connection holding time) as a function of the load. The

benchmark strategy, which employs the redundant resources only for failure recovery displays 100% availability. The lower blocking obtained by DP-PSSS comes at the expense of slightly degraded availability due to preemption of connections which traverse a backup SSS in the event of a working module failure. However, this degradation is relatively contained, as it is shown that the average value of the connection availability never drops below four nines.

5.3 Restoring Optical Cloud Services with Service Relocation

The optical cloud is a very popular concept by which storage and computing resources, also referred to as IT resources, are spread over different datacenter (DC) locations interconnected via high-speed optical wavelength division multiplexing (WDM) links. In this paradigm cloud services are provisioned in an anycast fashion, where only the source node needs to be specified in the routing and resource assignment phase, while any of the DCs can be used to accommodate a cloud service, as long as its IT resource requirements are met. Anycast provisioning has a number of advantages already recognized by optical cloud providers [36, 37]. It allows, for example, for the live relocation of those cloud services that are already provisioned, i.e., a concept also known as service relocation. If the DC location is not essential while provisioning a certain cloud service, it becomes then possible to relocate the job over multiple DC locations. This allows for a more efficient management of both cloud and transport network resources. There are also energy benefits that might derive from relocating, whenever possible, a service in a DC powered by green energy source [62].

Service relocation has also the potential to bring an extra degree of flexibility to survivability strategies. In fact, by opportunely relocating a cloud service it is possible to use a protection path terminating at a DC that is different from the one used by the primary path. These benefits have been assessed in the literature showing the ability of service relocation to improve resource efficiency when used in conjunction with path protection strategies [37]. Another instance in which service relocation might be beneficial is when it is combined with restoration-based survivability strategies. These strategies are very efficient in the way they use backup resources, i.e., they are dynamically provisioned only upon a failure, but restoration-based approaches suffer from a certain risk that the backup network resources might not be available when needed [17]. Service relocation can potentially alleviate this problem. Our preliminary study in this direction confirms that relocating a cloud service disrupted by a network failure is beneficial in terms of both restorability and average connection availability [19]. The results in [19] are based on the solution of an integer linear programming (ILP) model, which is fairly complex and does not scale well.

In order to fully assess the benefits of combining the cloud service relocation concept with restoration, we proposed, in addition to the ILP formulation [19], an efficient and scalable heuristic algorithm. A simple but powerful strategy was developed, able to jointly solve the restoration and service relocation problem. The proposed heuristic, called $H\_RELOCATION$ and presented in this section, selects for each failed cloud service the best combination of IT and network resources such that the number of restored cloud services is maximized, while the number of cloud service that need relocation is minimized. Simulation results show that the performance of $H\_RELOCATION$ in terms of both restorability and average connection availability is very close to the optimal result obtained by the ILP formulation used as a benchmark [19], while offering a significant gain in terms of processing time.

Input:
$G(N, E)$, N_{DC}, $Q_i \in Q$ and network state
Output:
selectedRoute if path is provisioned else NULL
Auxiliary:
curRoute := NULL

1: **if** ((selectedRoute := **shortestPath**(s_i, d_i)) != NULL)
  return selectedRoute;
2: **for** each $DC_k \in N_{DC} \mid (DC_k^{st} >= Q_i^{st}) \wedge (DC_k^{vm} >= Q_i^{vm})$
3: curRoute := **shortestPath**(s_i, DC_k);
4: **if** (**hopCount**(curRoute) < **hopCount**(selectedRoute))
  selectedRoute := curRoute;
5: **if** (selectedRoute = NULL) **return** NULL;
6: **else return** selectedRoute;

Figure 5.7: Pseudo-code for $H\_RELOCATION$ heuristic.

5.3.1 Restoration with Service Relocation Heuristic

This section first describes the restoration with service relocation problem and then presents a heuristic (i.e., $H\_RELOCATION$) specifically tailored to solve it. Let $G(N, E)$ be the graph describing the status of the optical transport network after the occurrence of a failure (i.e., where the failed network element(s) are not included in the graph representation). In this work only a single failure scenario is considered, but the problem and the proposed heuristic can be easily extended to a multi-failure scenario. $G(N, E)$ consists of N nodes and E fiber links. N_{DC} represents the set of datacenters nodes ($N_{DC} \subset N$) each one having ${DC_k}^{st}$ available storage units, and ${DC_k}^{pu}$ available processing units, with $DC_k \subset N_{DC}$. Let Q be the set of disrupted cloud services that need to be restored after the occurrence of a failure in the network, each one ($Q_i \subset Q$) requiring ${Q_i}^{st}$ storage units and ${Q_i}^{pu}$ processing units. The source node of Q_i is s_i, while d_i represents the DC node serving Q_i before the failure. The auxiliary variable $curRoute$ is used to store the temporary best route to a cloud service.

The objective of the restoration with service relocation problem is to maximize the number of recovered cloud services $Q_i \subset Q$ while minimizing the number of service relocations. The remainder of the section presents a heuristic which solves this problem.

The $H\_RELOCATION$ heuristic is described in Fig. 5.7. In the figure the **shortestPath** function returns the shortest available (i.e., with available wavelengths) path between two given nodes out of a set of pre-computed k shortest paths between them. If no such path exists or if there are no available wavelength resources on any of the pre-computed paths the **shortestPath** function returns NULL. The function **hopCount** returns the number of hops of a given route, or MAX_VALUE when curRoute=NULL.

$H\_RELOCATION$ tries to restore each cloud service $Q_i \subset Q$ sequentially. For each Q_i the heuristic first checks if there is an available path in $G(N, E)$ from s_i to the DC already in use (i.e., d_i). This is done to reduce the number of unnecessary service relocations. If no such path exists $H\_RELOCATION$ tries to find an alternative DC with enough storage (i.e.,${Q_i}^{st}$) and computing (i.e., ${Q_i}^{pu}$) resources able to accommodate Q_i. If more than one DC with enough resources is reachable, the heuristic chooses the one that is the closest to s_i in terms of hop count. Once the new DC is selected the cloud service is relocated

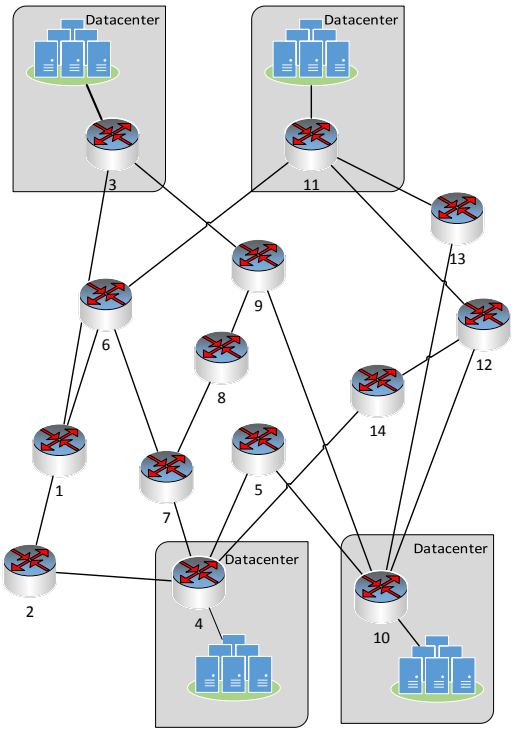


Figure 5.8: NSF topology with 4 DCs.

and a lightpath from s_i to the new DC is established. In case neither a suitable DC nor an available path to it can be found, the cloud service is dropped.

5.3.2 Performance Assessment

The NSF network (Fig. 5.8) is considered as a reference topology [19], where nodes 3, 4, 10 and 11 are assumed to be DC nodes due to their high connectivity. Each DC is equipped with 3000 storage units and 150 processing units. All fiber links in the network are bidirectional, comprising 16 wavelengths each (with the exception of the link connecting each DC to its network node, which have unlimited capacity). All nodes have full wavelength conversion capability.

Each experiment consists of one million anycast cloud services to be provisioned from a client (i.e., non-DC) node to a DC node (which are uniformly chosen for each cloud service) with enough storage and processing resources, in the interval from 1 to 100 and from 1 to 5, respectively. Each cloud service is also assumed to require the entire capacity of a wavelength channel.

In the normal operating conditions each cloud service is provisioned using the $DC\_CLOSEST$ heuristic [61], while upon the occurrence of a failure, $H\_RELOCATION$ is applied to restore as many cloud services as possible. Each cloud service holding time is assumed to be exponentially distributed with an average value equal to 60 time-units while service request arrivals at the client nodes follow a Poisson process, with mean time between arrivals defined by the current load. Only fiber link failures are considered in this work. The time between two consecutive failures in the network is exponentially distributed, with a mean value equal to 1000 time units, while the link reparation time, also exponential distributed, has a mean time to repair equal to 10 time units.

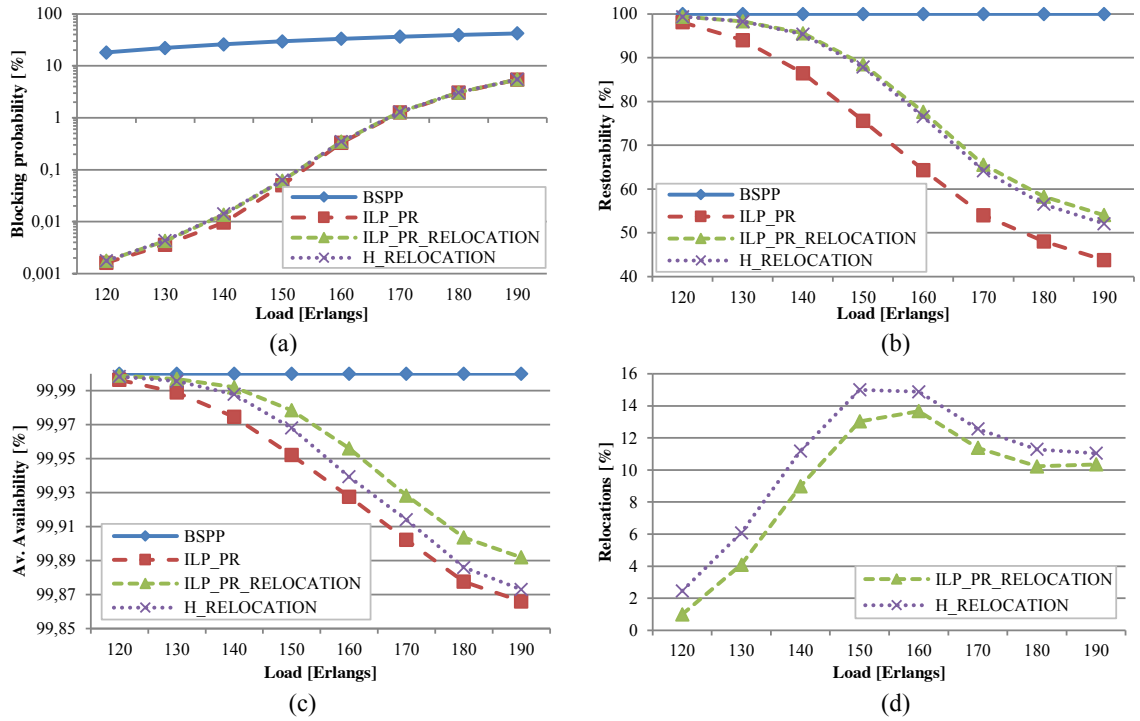


Figure 5.9: Simulation results.

All the presented results are the average of 30 different experiments, carried out using a Java-based discrete event-driven simulator [15]. The workstation used is a Red Hat Enterprise Linux with 8 Intel Xeon CPUs (4 cores per CPU) clocked at 2.67 GHz and with 16 GB of RAM memory. The confidence interval of the blocking probability is 5% or lower, with 95% of confidence level, except for the lower loads.

In the performance evaluation phase three survivability strategies are used for benchmarking purposes. The first one is a protection strategy for anycast cloud services called Backup Server via Physically disjoint Path (BSPP) [86], while the other two are ILP models representing: (i) an optimal restoration strategy without relocation capabilities (i.e., ILP_PR) [17], and (ii) our optimal restoration strategy with relocation capabilities and minimum number of cloud service relocations (i.e., $ILP\_RELOCATION$) [19].

Fig. 5.9 shows some performance results of the $H\_RELOCATION$ strategy as a function of the network load. The first thing that can be noticed is the inherent benefit, in terms of blocking probability (Fig. 5.9a), when using restoration-based strategies. More than one order of magnitude can be gained at low to medium loads when no dedicated protection resources are assigned. In addition it can also be noticed that using cloud service relocation (and consequently potentially longer restoration paths) has a little impact on the blocking probability. Fig. 5.9b shows the ratio between the number of successfully recovered cloud services over the total number of recovery attempts (i.e., restorability) as a function of load. As expected, BSPP guarantees 100% recovery against any single failure. However, as shown in Fig. 5.9a, this comes at the expense of very high blocking probability, even in relatively low load conditions. On the other hand, the relocation feature is quite efficient in increasing the chances of a cloud service being recovered. In this respect $H\_RELOCATION$ presents restorability values that are very close to the optimal (i.e., $ILP\_RELOCATION$). Similar conclusions can also be drawn while exam-

ining the average connection availability (i.e., the ratio between the time a cloud service is operative over the entire cloud service holding time) value shown in Fig 5.9c. In this case $H\_RELOCATION$ is slightly worse than $ILP\_RELOCATION$, which is due to the sequential nature of the heuristic that cannot concurrently optimize the restoration attempt of batch of disrupted cloud services. Finally, Fig 5.9.d shows efficiency of the proposed $H\_RELOCATION$ heuristic in containing the number of required relocations (i.e., ratio between the number of restored cloud service that required relocation over the total restoration attempts) when compared to their minimal value (i.e., $ILP_PR_RELOCATION$). The graph also shows that when the blocking probability is low, the number of relocations grows with the load, but after the system saturates, the number of relocations stabilizes, showing that the DCs and network resources are already highly utilized and it is difficult to accommodate services to be relocated from the affected DC to another DC, which reduces the number of possible relocations when the load is further increasing.

5.4 Conclusions

This chapter first addressed the problem of guaranteeing high survivability levels in WDM transport networks in the presence of multiple failures. More specifically, the work focused on double link failure scenarios and proposes two hybrid survivability schemes, namely DPP + PR and DPP + BR + PR. These schemes combine the backup reprovisionig (BR) concept with an end-to-end path restoration (PR) scheme with a focus on maximizing the connection availability without a significant impact on the blocking performance. These two schemes were evaluated against two benchmark solutions, namely DPP (1:2) and DPP + BR. Simulation results show that both proposed schemes achieve substantially better blocking probability performance than DPP (1:2), while still maintaining acceptable connection availability levels. Furthermore, their performance in terms of connection availability is far better than DPP+BR. Finally, the DPP + BR + PR scheme results in low connection downtime values, and drops only half as many connections as DPP + PR under high load, which is an important performance parameter for network service providers.

As a second contribution the chapter presented a dynamic connection provisioning strategy for optical networks with programmable ROADMs. The proposed strategy exploits the presence of switching modules deployed for failure recovery to establish connections, which would otherwise be dropped. Simulation results confirm that the proposed approach reduces connection blocking by more than 50% without a drastic impact on the connection availability performance.

Finally the chapter presented a sequential heuristic (i.e., $H\_RELOCATION$) aimed at restoring a set of failed cloud services using service relocation in a dynamic optical cloud network scenario. When benchmarked against optimal solutions provided by two ILP formulations, $H\_RELOCATION$ shows restorability and average connection availability values that are very close to the optimum, with only a slightly increase in the number of services that need to be relocated to a different datacenter.

Chapter 6

Survivable Optical Metro/Core Networks with Dual-Homed Access: an Availability vs. Cost Assessment Study

Long Reach Passive Optical Networks (LR-PONs) allow to consolidate the number of electronic nodes in the central office (CO) and to provide maximum reach of the access segment in the order of a hundred kilometers, while being able to support a large number of customers in the service area. Nonetheless, certain failures, especially in the feeder part of a LR-PON, may affect a large number of customers. One possibility to protect customers from being disconnected after a feeder fiber cut or a CO failure is to use dual-homing.

As can be seen from Fig. 6.1, 100% survivability for a LE-to-LE connection under any single M/C node failure scenario can be ensured by provisioning a pair of node-disjoint paths i.e., applying dedicated-path protection (DPP). However, DPP may be quite expensive in terms of resources to be deployed and reserved in the M/C nodes, i.e., both the number of OLTs on the client-side [69] and WDM transponders on the line-side, where lightpaths are initiated/terminated.

In the work presented in this chapter, we propose a different approach to providing survivability for dual-homed LE-to-LE connections in the presence of single M/C node failures. Our hypothesis is that multilayer restoration in combination with an adequate over dimensioning of the number of WDM transponders in the M/C nodes can yield similar connection availability performance as DPP, but at a significantly smaller resource overbuild (i.e., number of WDM transponders deployed in each M/C node). This proposal is based on the reasoning that allocating extra transponders in M/C nodes increases the chances of finding backup paths during the restoration process. In this context, the following question arises: how much overprovisioning (i.e., extra WDM transponders) is needed to obtain a favorable tradeoff with connection survivability performance?

In order to answer this question we studied two design strategies that can be used to decide what is a reasonable level of overprovisioning for the WDM transponders. The strategies work in a similar way. They both begin by evaluating the number of extra transpon-

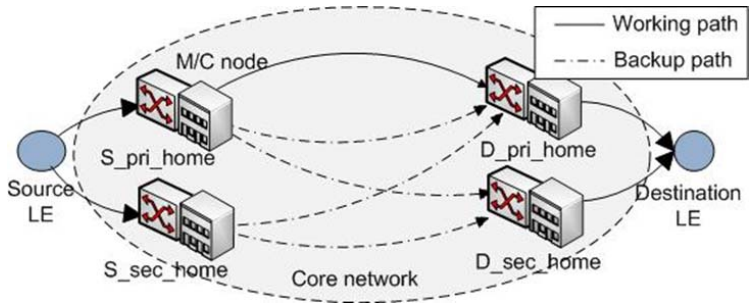


Figure 6.1: Network architecture with dual-homed local exchanges (LEs) with a working path and different options for backup paths.

ders needed at each node for ensuring successful restoration of all connections disrupted in each possible single M/C node failure scenario. The first strategy, denoted as Dual-Homing Path Restoration Max (DH-PR-MAX), then allocates in each M/C node the number of extra transponders needed to recover from the M/C node failure scenario which disrupts the maximum number of connections. The second strategy, denoted as Dual-Homing Path Restoration Average (DH-PR-AVG), allocates the average number of extra transponders needed for recovering connections from M/C node failure calculated over all M/C node failure scenarios. For benchmarking purposes, we propose also a multilayer design algorithm to provide DPP, i.e., Dual-Homing DPP (DH-DPP). This strategy is extended from the algorithm proposed in [14] to support the wavelength continuity constraint.

The performance of the two proposed design strategies in terms of the number of deployed transponders is compared with DH-DPP in a realistic network dimensioning scenario. The three design strategies (i.e., DH-PR-MAX, DH-PR-AVG, and DH-DPP) are then deployed in a live network operation scenario where LE-to-LE connections and single M/C node failures appear at random, and evaluated in terms of average connection blocking probability and connection availability performance. Our results demonstrate that by combining multilayer restoration with the right amount of transponder overprovisioning it is possible to achieve average connection availability very similar to DPP but with lower blocking, and using a lower (up to 35%) number of WDM transponders. The results were obtained using the open-source Net2Plan tool [74], and source code is available on the website [2].

6.1 Reference Architecture

Without loss of generality, we consider an IP/MPLS-over-WDM multilayer architecture where LEs represent source and destination of the IP traffic. M/C nodes are in charge of routing the traffic on top of a full-mesh of lightpaths, built over an arbitrary fiber topology interconnecting such nodes.

In our model, M/C nodes are equipped with an optical switch (i.e., reconfigurable optical add/drop multiplexer) co-located with an IP/MPLS router, which aggregates traffic from/to LEs. Transponders installed between IP/MPLS routers and optical switches are used to initiate/terminate lightpaths. Installed transponders are assumed to be tunable in each direction. In addition, M/C nodes acting as homes (Fig. 6.1) are the only electronic processing interface between source and destination LEs, and in order to better utilize lightpath capacity, low rate traffic is multiplexed into lightpaths in the source M/C node and demultiplexed in the destination M/C node.

The design algorithms will be in charge of determining how many transponders have to be installed in each M/C node to support the forecasted traffic, that is, to establish the required number of lightpaths and route the traffic on top of them, when the network is failure-free. Additionally, algorithms should also consider that some failures may happen, and install extra transponders for backup lightpaths.

Regarding the dynamic scenario, which is used as an assessment framework for blocking and availability performance comparison in the second part of our study, we consider that the network is orchestrated by a centralized controller with full information about the network state.

This element receives notification of different events (i.e., connection requests, failure notifications) and is taking decisions (i.e., set-up a new lightpath, reroute traffic) according

Input: Set of LEs and M/C nodes, LE-MC dual-home assignment, fiber topology, traffic matrix

Output: Number of transponders to be installed in each M/C node

Step 1. Sort demands from traffic matrix according to a certain criteria.

Step 2. For each demand, apply the VTD+RWA allocation algorithm corresponding to the given resilience scheme.

Step 3: For each M/C node, compute the number of transponders to be installed as the maximum between the number of incoming and outgoing lightpaths from/to that node.

Figure 6.2: Scheme of the network design algorithms.

to a control and management plane module that is disseminated to distributed agents installed in M/C nodes across the network. Protocols and interaction mechanisms between the controller and the rest of the elements are out of the scope of the section.

6.2 Network Design And Control Plane Algorithms

In this section, we provide a high-level description of the developed algorithms. Due to the space limitation we do not show all the implementation details, but the source code can be found in [2].

Our design algorithms solve both the virtual topology design (VTD) problem and the routing and wavelength assignment (RWA) problem. The former decides how many lightpaths to install, whereas the latter is devoted to determine their physical route and wavelength. Inputs of our algorithms are the set of LEs and M/C nodes, the LE-MC dual-home assignment, the fiber topology, and a traffic matrix, which contains aggregated information of the traffic offered by each LE-LE pair. The output of the algorithm gives the total number of transponders to be installed in each M/C node, our cost results, calculated as the maximum between the number of incoming and outgoing lightpaths per M/C node. Recall that we assume our transponders are tunable independently in each direction (transmission and reception).

It is well-known that the combined VTD+RWA problem is computationally intractable. Hence, our algorithms are based on the following heuristic scheme: (i) we sort the set of static demands from the traffic matrix according to a certain criteria, and (ii) we accommodate them on a one-by-one basis, using a certain allocation algorithm. Algorithm 1 in Fig. 6.2 summarizes the skeleton of the design algorithms.

Interestingly, this scheme is readily adaptable to the dynamic scenario, where connection requests arrive independently, since the allocation algorithm can be reused. The extended algorithm for the in-operation scenario should also include three additional functionalities, each of them related to one of the new possible events received by the network controller: (i) connection release, (ii) failure detection, and (iii) reparation notification. Naturally, the allocation algorithm of the design phase is applied upon arrival of connection request events.

In the following subsections, we describe the pseudo-code of the proposed approaches. To simplify the description, for each resilience scheme (protection and restoration) we first present the allocation algorithm (called “Connection request”), applied either in the design phase or in the in-operation phase, and then the other three reaction-to-event procedures.

To avoid ambiguities and for clarity of the description, we remark that whenever we mention “try to set-up lightpath” and “find a lightpath with spare capacity” actions, we refer to the

Step 1. If LEs share the same M/C node assignment (either primary/primary + secondary/secondary or primary/secondary + secondary/primary), go to Step 6. Otherwise, go to Step 2.

Step 2. If LEs share one M/C home node (any combination: primary/primary, primary/secondary, secondary/primary, secondary/secondary), only a node-disjoint lightpath (using the other M/C home node) is required.

Step 2.1. Find a node-disjoint lightpath (using the other M/C home nodes) with spare capacity. If found, go to Step 6. Otherwise, go to Step 2.2.

Step 2.2. Try to set-up a new node-disjoint lightpath (using the other M/C home nodes). If established, go to Step 6. Otherwise, go to Step 3.

Step 3. Try to find an existing node-disjoint lightpath pair with spare capacity. If found, go to Step 6. Otherwise, go to Step 4.

Step 4. Try to find a lightpath with spare capacity, and set-up an additional node-disjoint lightpath. If possible, go to Step 6. Otherwise, go to Step 5.

Step 5. Try to set-up a node-disjoint lightpath pair using the following adaptation of the wavelength-scan algorithm in [10].

Step 5.1. For each wavelength, run Suurballe's algorithm and return the two disjoint paths with the minimum total length. If found, go to Step 6. Otherwise, go to Step 5.2.

Step 5.2. For all wavelengths, run Dijkstra's algorithm to find a lightpath in each of them, and loop over all the wavelengths again to find a node-disjoint lightpath with the minimum total length. If found, go to Step 6. Otherwise, block the request.

Step 6. Assign resources to allocate the connection.

Figure 6.3: Connection request procedure for the DH-DPP algorithm.

set of resources that are currently in the operational state, that is, not affected by any failure. Also as a general rule, to set-up a new lightpath, all the wavelength planes are explored, taking the shortest path (in km) among all candidates, using first-fit wavelength assignment. Note that the number of lightpaths that can be established between each M/C node pair is limited by the number of installed transponders. On the other hand, since IP/MPLS routes only traverse at most one lightpath, IP traffic is routed using best-fit, that is, we choose the lightpath whose spare capacity is the closest to the requested bandwidth of the connections. Ties are broken arbitrarily.

6.2.1 Dual-homing dedicated path protection algorithm (DH-DPP)

The first algorithm is used for 1:1 node-disjoint protection, which guarantees 100% availability under single M/C node failure by allocating each connection over a pair of node-disjoint lightpaths. If such pair of lightpaths cannot be found, the connection is blocked.

Before describing the allocation algorithm, we would like to remark that Suurballe's algorithm [87] is the reference algorithm for finding pairs of node-disjoint paths in IP/MPLS networks. However, the problem is challenging (in fact NP-complete [22]) in the context of non-wavelength-convertible IP/MPLS-over-WDM networks due to the additional wavelength conversion constraint. For this reason we decided to use a modified version of the algorithm presented in [97], which first tries to apply Suurballe's algorithm on each wavelength plane, and if that fails, applies Dijkstra's algorithm on different wavelength planes.

Connection request allocation algorithm. When a request is received, the algorithm explores all possible options to allocate the connection. To reduce cost, we try to reuse resources on existing ligthpaths as much as possible, while we try to set-up new lightpaths only if we cannot find a node-disjoint pair of existing lightpaths with spare capacity. The pseudo-code of the procedure is shown in Fig. 6.3.

Step 1. Find a lightpath with spare capacity. If found, go to Step 3. Otherwise, go to Step 2.

Step 2. Try to set-up a new lightpath. If established, go to Step 3. Otherwise, block the request.

Step 3. Assign resources to allocate the connection.

Figure 6.4: Connection request procedure for the DH-PR algorithm.

Connection release. Whenever a connection is released, the related lightpath pair capacity is released. If this leaves some lightpath empty, it is also torn-down and the associated transponders are released.

Failure detection and reparation notification. Upon a failure, affected connections are transferred to their pre-reserved backup lightpaths. Upon reparation, they are restored back to their original working lightpaths.

6.2.2 Dual-homing with path restoration (DH-PR)

This strategy considers that the network is able to set-up backup lightpaths on demand after a failure occurs, and provisions only working paths in the design phase. In our case study, we analyze two variants of this algorithm. In the first variant, denoted as "usePrimaryMC", we assume that connections can only be accommodated by primary MC-to-primary MC lightpaths. In the second variant, denoted as "useAnyMC", we consider that any of the four possible path combinations can accommodate connections. The decision on the applied variant can be made by the operator depending on the technological constraints.

The design algorithm depicted in Fig. 6.2 is used to calculate the number of transponders for the unprotected (failure-free) scenario, denoted as "DP-PR-UNP". To ensure additional resources for connection restoration in case of failures, DH-PR needs to be combined with different over-dimensioning strategies. As discussed in [92], the number of extra transponders can be calculated as the worst case among all possible single M/C node failures (the DH-PR-MAX approach). However, due to the very low utilization of some lightpaths (below 10%), reasonable availability figures could still be provided at a reduced cost. Therefore, in the DH-PR-AVG scenario, nodes are equipped with the average number of extra transponders required across all possible failure scenarios.

Connection request allocation algorithm. The process entails provisioning only the working path of the request. The pseudo-code of this procedure is shown in Fig. 6.4.

Connection release. Whenever a connection is released, the related lightpath capacity is released. If this leaves the lightpath empty, it is also torn-down and the associated transponders are released.

Failure detection. Upon a failure, the algorithm considers a bottom-up escalation strategy [92] based on a refinement of the optical-followed-by-IP restoration mechanism presented in [54]. The algorithm first tries to reroute the failing lightpaths over the surviving fiber topology. For restored lightpaths, the IP layer remains unnoticed. For non-restored lightpaths, their carried connections are rerouted using the allocation algorithm (Fig. 6.4). If a connection cannot be rerouted, it goes down until resources become available. Note that, by recovering first at the optical layer only one event has to be treated and the number of required recovery actions is minimal, compared to rerouting multiple connections at the IP layer.

For the “usePrimaryMC” scenario, we always try to reroute the traffic over primary-to-primary lightpaths, and if and only if primary-to-primary lightpath cannot be found or set-up, the traffic is switched to any of the other possible combinations. Hence, secondary homes are used only when primary ones cannot provide backup paths.

Reparation notification. Upon reparation, affected lightpaths and connections are restored to primary paths using a make-before-break policy [88].

6.3 Case study

In this work, we analyze a simple scenario where only one single M/C node may fail at a time. Depending on the design strategy, we dimension the number of transponders that should be installed in the M/C nodes accordingly. The case study is divided into a dimensioning and in-operation phase.

Dimensioning phase. In this first stage, we start with an empty multilayer network design and a set of LE-LE traffic demands from traffic forecasts. The network design algorithms are fed these traffic demands sorted in descending order of the requested bandwidth and physical distance product, so that priority is given to demands requesting high capacity over long distances.

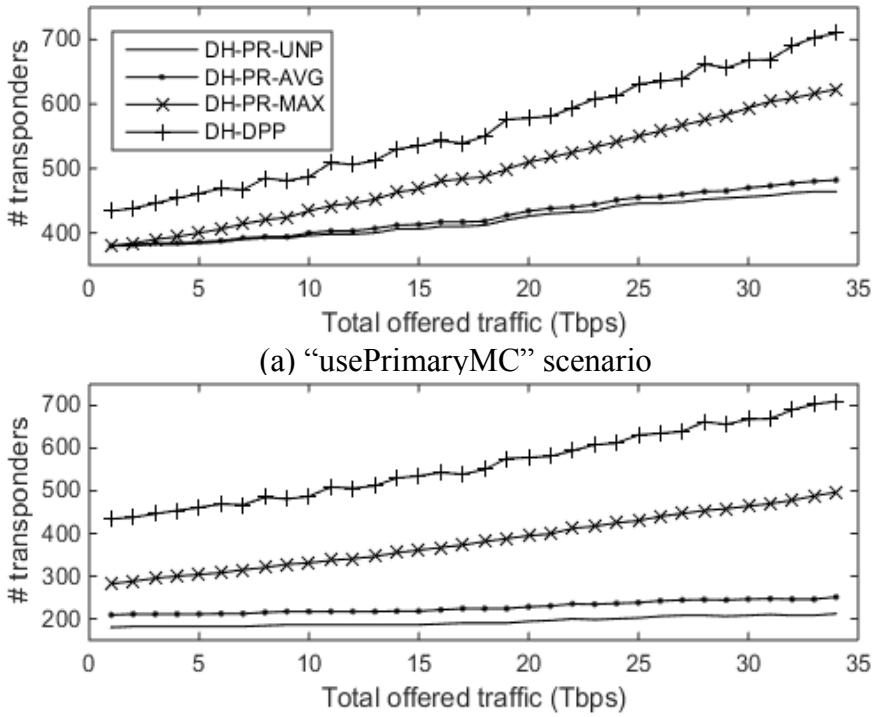
In-operation simulation. In the second stage of the study, we take as an input the same empty network design used for the dimensioning, but using the number of transponders obtained after running the first phase. We then perform a long-run simulation of network operation, where connections and single M/C node failures appear at random. In each scenario, we evaluate blocking probability and availability performance. The former refers to the quotient between the number of offered and accepted connections. The latter represents the average traffic survivability, which is equal to the average ratio of the up time over the holding time for the set of accepted connections.

6.4 Results

This section presents an analysis of the resource consumption (in terms of number of transponders) and blocking/availability metrics obtained by the proposed algorithms in the case study just described.

In our tests, we consider a country-wide network topology consisting of 1204 LEs and 20 M/C nodes. Information provided by the local operator includes only fiber links (supporting up to 40 wavelengths [88] of 100 Gbps of capacity) between M/C nodes. We assume that the M/C node closest to each LE is its primary home, and the second closest one is its secondary home.

The traffic matrix between LE pairs is obtained from a population-distance model, where traffic T_{AB} between LE A and B is calculated as: $T_{AB} = K \times (N_A \times N_B)/D^2$. N_A and N_B represent the number of users served by A and B, respectively, D is the distance between the two LEs, and K is the traffic load factor. We use different scaling factors K to model different total offered traffic values.



(a) "usePrimaryMC" scenario

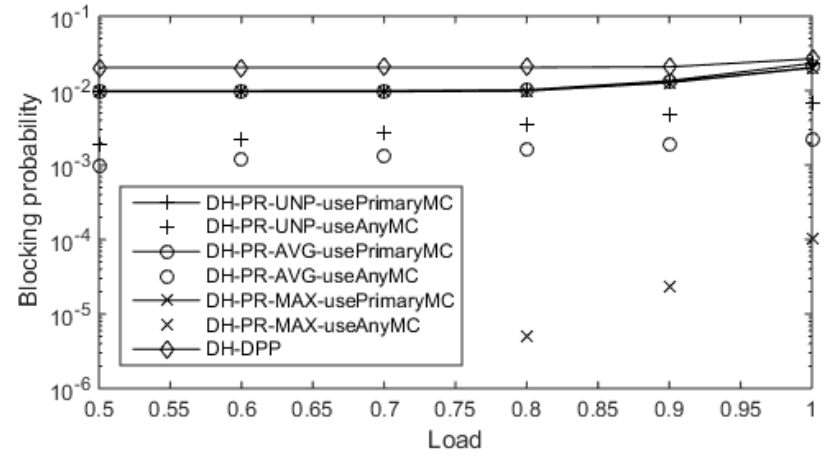
(b) "useAnyMC" scenario

Figure 6.5: Total number of transponders required by each dimensioning strategy.

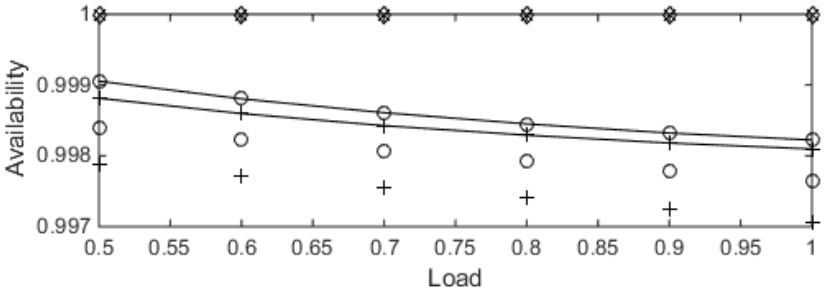
6.4.1 Dimensioning phase

We executed our design algorithms for 34 growing traffic matrices, ranging from a low traffic scenario with 1 Tbps of total offered traffic to a high traffic load with 34 Tbps of total offered traffic. We observed that with 35 Tbps of offered traffic, we could not provision enough resources for all the traffic.

Figure 6.5 shows the total number of transponders to be allocated in the M/C nodes for each design strategy. As can be seen, trends are similar for both "usePrimaryMC" (Fig. 6.5a) and "useAnyMC" scenarios (Fig. 6.5b). As expected, DH-PR-MAX is a cheaper alternative to DH-DPP considering that both provide 100% availability under single M/C node failure in the design phase, and similar behavior is expected in the in-operation phase. The number of transponders is reduced by 15% and 35% for the "usePrimaryMC" and "useAnyMC" scenarios, respectively. Besides, there is a huge gap between DH-PR-MAX and DH-PR-UNP/AVG. This is due to the fact that there is a single M/C node failure that dramatically affects the traffic survivability over the whole set of failures. Recall that if LEs share an M/C home node (the primary home, in case of "usePrimaryMC", or any in case of "useAnyMC"), connections between them do not require lightpaths as they do not enter the core in a failure-free state. In the used topology, we found that there is a LE pair whose offered traffic accounts for 35% of the total offered traffic, and shares the primary M/C home. Hence, traffic only enters the M/C network in case of failure, requiring a lot of additional lightpaths to be established. Differences between "usePrimaryMC" and "useAnyMC" scenarios come from the following observation: in "usePrimaryMC" scenario, we enforce traffic to use primary-to-primary lightpaths (except when LEs share primary homes), even though LEs may share some other M/C node (any possible combination but primary-to-primary). Hence, the number of lightpaths for the unprotected design is larger



(a) Connection blocking probability



(b) Connection availability

Figure 6.6: Performance metrics for the in-operation simulation.

than for "useAnyMC". In contrast, many additional lightpaths (in relative numbers) are required for the "useAnyMC" approach with respect to "usePrimaryMC" in case of failure, since all the traffic carried in the network edge for the unprotected case will now enter the core network. Note that results for DH-DPP do not depend on the scenario because both scenarios apply the same allocation algorithm.

6.4.2 In-operation simulation

The in-operation simulation allows us to assess the performance of the algorithms in terms of connection blocking probability and connection availability. In our experiments, the connection inter arrival time and connection holding time are assumed to be exponentially distributed and independent for each LE-LE pair, where the connection requested bandwidth is equal to 1 Gbps. The mean inter arrival time is set to 1 time unit. Holding times are adjusted to match a certain traffic load, proportional for each LE-LE pair to its value in the traffic matrix.

Similarly, we assume that the mean times to failure and to reparation of M/C node are also exponentially distributed, with mean average values equal to 0.9 and 0.1units, respectively. The failed M/C node is chosen randomly using a uniform distribution. Note that the failure arrival rate is chosen to be much more aggressive than in reality to test the robustness of our approaches as well as to obtain statistically meaningful performance results within a reasonable amount of simulation time. For each scenario, we run the simulation with 10^8 connection requests (10% of transitory).

Figure 6.6 shows the results for the two performance metrics for a single design scenario with 30 Tbps of total offered traffic. The results for other traffic values are similar and omitted due to lack of space. The holding times are adjusted to match certain loads values (or fractions of 30 Tbps).

Fig. 6.6a represents the blocking probability. As can be seen, the DH-DPP algorithm performs the poorest. This is due to the fact that a connection cannot be allocated unless two active node-disjoint lightpaths are found. The DH-PR-UNP and DH-PR-AVG approaches also obtain high blocking probability. Interestingly, the blocking probability of DH-PR-MAX approach is relatively poor when it is combined with "usePrimaryMC", but improves drastically when the "useAnyMC" strategy is applied, remaining below 10^{-4} for all considered loads. Note that the gap between the "usePrimaryMC" and "useAnyMC" strategies stems from the allocation algorithm definitions. Namely, while "useAnyMC" considers up to 4 possible combinations for the primary path, the "usePrimaryMC" case enforces the usage of primary M/C home nodes for the primary path and blocks the connections otherwise.

Fig. 6.6b shows the average connection availability. Apart from DH-DPP, which guarantees 100% availability in the considered single-node failure scenario, DH-PR-MAX achieves more than six-nine availability at any load, indicating that the suitable degree of overdimensioning has been found. Interestingly, unavailability in the case of "useAnyMC" is higher than in "usePrimaryMC". This can be explained by the fact that the higher the number of active connections (Fig. 6.6a) implies lower spare capacity to reroute traffic in case of failure. Again, performance of DH-PR-UNP and DH-PR-AVG is poor even at low loads, which can be expected considering their low number of allocated transponders.

6.5 Conclusions

In this work, we present a study on the dimensioning of metro/core networks with dual-homed access under the assumption of single metro/core node failure scenario. Our results show that by equipping each M/C node with just enough transponders to recover from the most-damaging M/C node failure scenario and by applying path restoration, it is possible to obtain almost the same availability performance of dedicated path protection but at a significantly reduced cost (up to 35% less transponders).

Chapter 7

Impact of Energy-Efficient Techniques on a Device Lifetime

Telecom operators became particularly interested in energy-efficient techniques, since they see green ICT as a possible way to reduce their electricity bill. Looking at backbone networks, the active network elements consume a non-negligible amount of energy [91]. This is due to the fact that they have to handle a huge amount of traffic. This requires high-capacity devices which are constantly powered on a 24-hour basis. Moreover, the traffic exchanged by users will continuously increase in the forthcoming years, due to the penetration of Internet connectivity as well as the multitude of smart devices coupled with bandwidth-intensive services. As a result, the power consumption of backbone networks is expected to continue growing [52], unless proper counter measures are taken into account.

To this end, different energy-efficient approaches have been proposed in the literature, targeting both design and dynamic operations of backbone networks (see [28, 38] for detailed surveys). Focusing on optical backbone networks, the majority of the energy-efficient algorithms are based on putting optical devices (e.g., transponders, regenerators, Reconfigurable Optical Add/Drop Multiplexer (ROADMs), and Erbium Doped Fiber Amplifiers (EDFAs)) into an energy-efficient mode (e.g., *sleep* mode) to reduce their power consumption. However, a possible drawback of this approach is that frequent on/sleep switching may negatively impact the failure rate performance of a device, and consequently increase its reparation costs [29, 30, 94]. This aspect cannot be neglected when considering the overall benefits of a green strategy. In particular, it is important to make sure that the potential savings brought by a reduced power consumption level are not lower than the possible extra reparation costs caused by a reduced lifetime. The work presented in Deliverable 7.1 [9] pointed out this aspect and assessed different types of active components with the maximum allowable failure rate increase due to setting them into sleep mode on a regular basis to save energy.

Leveraging these previous works, the aim of the work presented in this chapter is to provide answers to the following questions: *(i)* what are the main sleep mode and hardware (HW) parameters that influence the failure rate performance of EDFAs in optical backbone networks?, and *(ii)* under which conditions may the lifetime of an EDFA increase as a result of being set into sleep mode? The material presented in the chapter is organized as follows. We first present the main causes that impact the EDFA lifetime. Then, a failure rate acceleration factor model is presented in order to understand the role of both sleep mode and HW parameters in the potential changes of the failure rate of an EDFA . The model is then evaluated by considering energy-efficient schemes that are based on putting the EDFA into sleep mode. In particular, the section considers an energy-efficient RWA algorithm, where the main idea is to route the traffic over already used fiber links. In this way it is possible to maximize the number of EDFAs that can be set into sleep mode and therefore save energy.

7.1 Impact of Sleep Mode Operations on a Device Lifetime

This section provides a general overview of the physical phenomena that may impact the lifetime of a network device put into sleep mode in order to save energy.

Transitions between *on* and *sleep* states affect the conditions in which a device operates, in particular its temperature [25]. There are several models that describe how temperature impacts the lifetime of a device. One of them is the Arrhenius law [24], which defines in terms of an acceleration factor how much the lifetime of a device could increase/decrease if operated at a temperature different from a reference one. More formally, the acceleration factor derived from the Arrhenius law can be expressed as:

$$AF^{\mathcal{T}_1} = \frac{\gamma^{\mathcal{T}_1}}{\gamma^{\mathcal{T}_r}} = e^{-\frac{E_a}{K}(\frac{1}{\mathcal{T}_1}-\frac{1}{\mathcal{T}_r})} \tag{7.1}$$

where E_a is the activation energy, K is the Boltzmann constant, and $\gamma^{\mathcal{T}_1}$ and $\gamma^{\mathcal{T}_r}$ are the failure rate at the operating $\mathcal{T}_1$ and reference $\mathcal{T}_r$ temperatures, respectively. If the effects of the Arrhenius law were the only phenomenon to consider, an energy-efficient scheme would have a positive impact on the lifetime of a device, as the operating temperature of a device in sleep mode is typically lower than in normal working conditions (i.e., when a device is powered on).

However, there are also other physical phenomena that need to be considered, which might negatively impact the lifetime of a device. For example, it is well known that temperature changes may affect differently the expansion of different materials within the same component due to different Coefficients of Temperature expansion (CTEs). In turn, the device may suffer strain and fatigue when temperature conditions change, in particular when this happens in a cyclic way. This phenomenon can be observed for many electronic devices, in particular in solder junctions. The Coffin-Manson model [34, 63] describes the effects of material fatigue caused by cyclic thermal stress and it is used to predict the number of temperature cycles that a component can endure before failing. More formally, the number of cycles to failure can be expressed by the following equation [35, 55]:

$$N^f = C_0(\Delta_{\mathcal{T}} - \Delta_{\mathcal{T}_0})^{-q} \tag{7.2}$$

where $\Delta_{\mathcal{T}}$ is the temperature variation, $\Delta_{\mathcal{T}_0}$ is the maximum temperature variation that can be afforded without an impact on the failure rate, C_0 is a material dependent constant, and q is the Coffin-Manson exponent. Then using Eq. (7.2), the failure rate due to thermal cycling can be defined as:

$$\gamma^{\Delta_{\mathcal{T}}} = \frac{f^{TC}}{N^f} \tag{7.3}$$

where f^{TC} is the frequency of thermal cycling and $\gamma^{\Delta_{\mathcal{T}}}$ is the estimated failure rate. The value of f^{TC} can be obtained via experimental measures. In this case, both the frequency of temperature changes and the temperature variation play an important role. It is because the more often a device experiences a temperature gradient (i.e., as a result of being put into sleep mode), the shorter its lifetime might be. There are other models available in the literature (e.g., Engelmeier [43], Norris-Lanzberg [70]) that introduce additional factors (e.g., solder dimensions, chemical characteristics, dwell time) into a lifetime prediction model. On the other hand, they all share their dependence on the temperature variation and frequency of the temperature cycles.

One could argue that on/sleep switching (i.e., power cycling) based schemes produce a localized and internally induced heating in the device (Joule heating) so that the resulting temperature variation may not be uniformly distributed over the whole device as it happens with thermal cycling, where the device heating is externally induced. On the other hand, the work in [78] and [76] confirms that the fast local temperature changes caused by power cycling (that can be up to 100 times faster [23] than the thermal cycling) also negatively impact the lifetime of a device. From the consideration above it can be concluded that temperature and the temperature variations are relevant phenomena that might impact the lifetime of a device.

When looking at optical backbone networks, the set of elements that may be the target of an energy-efficient scheme include: transponders, regenerators, ROADMs, and EDFAs. Among these various elements, the EDFA is the one where the impact of the temperature conditions and temperature variations is the most critical [94].

7.2 EDFA Failure Rate Model and Average Failure Rate Acceleration Factor

This section focuses on modeling the lifetime variations of an EDFA as a function of its temperature and temperature variations[1].

Let us denote as D_{ijk} the lifetime of the EDFA i placed on the fiber link from node j to node k. D_{ijk} is defined as the inverse of the EDFA failure rate γ_{ijk}:

$$D_{ijk} = \frac{1}{\gamma_{ijk}} \text{[h]}. \tag{7.4}$$

When an EDFA is in a low-power state (or sleep state), its temperature is reduced compared to the full-power state. According to the Arrhenius law, Eq. (7.1), this induces a decrease in the failure rate compared to the full power state. The failure rate $\gamma_{ijk}^{on-sleep}$ of EDFA i implementing sleep mode capabilities and placed on the fiber link between node j and node k is:

$$\gamma_{ijk}^{on-sleep} = \left[(1 - \tau_{ijk}^{sleep})\gamma_{ijk}^{on} + \tau_{ijk}^{sleep}\gamma_{ijk}^{sleep}\right] \text{[1/h]}, \tag{7.5}$$

where, γ_{ijk}^{sleep} is the failure rate when EDFA i is in sleep mode, γ_{ijk}^{on} is the failure rate at full power, and $\tau_{ijk}^{sleep} \in [0, 1]$ is the normalized time in which EDFA i is in sleep mode (if equal to 1 the EDFA i is always in sleep mode). Thus, the overall failure rate is the averaged sum of the failure rates at full power and in sleep mode. The first consideration is that the longer the EDFA is in sleep mode, the higher the reduction in the failure rate. However, as reported in Eqs. (7.2) and (7.3), also the interval in which the temperature varies (i.e., $\Delta_{\mathcal{T}}$) and the frequency of thermal cycling (i.e., f^{TC} in [cycle/h]) impacts the failure rate. Let us define γ_{ijk}^{tr} as the failure rate of the device when a temperature variation occurs. Following Eq. (7.3), γ_{ijk}^{tr} can be defined as:

$$\gamma_{ijk}^{tr} = \frac{f_{ijk}^{tr}}{N_{ijk}^{f}} \text{[1/h]}, \tag{7.6}$$

[1] The model is general enough that can be extended to model the failure rate and the failure rate variations of any other network device.

where f_{ijk}^{tr} is the on/sleep frequency and N_{ijk}^{f} is the number of cycles to failure. From Eq. (7.6) it can be seen that as the power switching frequency is reduced, the failure rate γ_{ijk}^{tr} is also reduced. Moreover, N_{ijk}^{f} is a technological parameter that depends on the specific HW used to assembly the EDFA.

In order to put together both the effects of Eq. (7.6) and Eq. (7.5), we assume that $\gamma_{ijk}^{on-sleep}$ and γ_{ijk}^{tr} are statistically independent [27] and their effects are additive [35]. In this way, the overall failure rate γ_{ijk} is the sum of the individual failure rates:

$$\gamma_{ijk} = \gamma_{ijk}^{on-sleep} + \gamma_{ijk}^{tr}[1/h]. \tag{7.7}$$

Since we are interested in evaluating in which way the sleep mode may impact the lifetime of an EDFA, we define a failure rate acceleration factor, i.e., AF_{ijk}, similar in concept to the one defined by the Arrhenius law, i.e., Eq. (7.1). The acceleration factor is a metric that measures the increase of the failure rate with respect to a reference temperature. In order to model the value of AF_{ijk}, we first define the failure rate acceleration factor of an EDFA in sleep mode as:

$$AF_{ijk}^{sleep} = \frac{\gamma_{ijk}^{sleep}}{\gamma_{ijk}^{on}}, \tag{7.8}$$

which is always lower than one. This term can be computed from using the Arrhenius law given the difference in the operating temperature of the device and its activation energy. Moreover, we introduce the parameter χ_{ijk}:

$$\chi_{ijk} = \frac{1}{\gamma_{ijk}^{on} N_{ijk}^{f}}[\text{h/cycle}], \tag{7.9}$$

which is defined as the inverse of the failure rate at full power multiplied by the number of cycles to failure. Both terms are fixed and can be measured on the device when sleep modes are not applied. Finally, the overall failure rate acceleration factor of an EDFA can be defined as:

$$AF_{ijk} = \frac{\gamma_{ijk}}{\gamma_{ijk}^{on}} = 1 - (1 - AF_{ijk}^{sleep})\tau_{ijk}^{sleep} + \chi_{ijk} f_{ijk}^{tr}. \tag{7.10}$$

The acceleration factor AF_{ijk} comprises two terms: the first one is $(1 - AF_{ijk}^{sleep})\tau_{ijk}^{sleep}$ which tends to decrease the AF_{ijk} value, and the second one $\chi_{ijk} f_{ijk}^{tr}$ which has the opposite effect. Moreover, AF_{ijk} is influenced by two types of parameters: technological (i.e., AF_{ijk}^{sleep} and χ_{ijk}) which are strictly related to the HW used to build the EDFA, and sleep-mode-related (i.e., τ_{ijk}^{sleep} and f_{ijk}^{tr}) which instead depend on the energy-efficient algorithm used.

When, in a given network, a number of EDFAs are put into sleep mode in order to save energy (with on/sleep switching frequencies and sleep periods that might not be necessarily the same for each one of them), it is important to model their overall performance in terms of failure rate acceleration factor. We define the following metric for this purpose:

$$\overline{AF} = \frac{\sum_{ijk} AF_{ijk}}{I}. \tag{7.11}$$

The $\overline{AF}$ in Eq. (7.11) defines the average EDFA failure rate acceleration factor, where I is the total number of EDFAs deployed in the network. If $\overline{AF} > 1$, it can be expected that, on average, the EDFAs in the network will fail more frequently than in normal operating

conditions (i.e., when a green strategy is not used). If $\overline{AF} = 1$, the average failure rate of EDFAs will not change, and if $\overline{AF} < 1$ it can be expected that, on average, the EDFAs in the network will fail less frequently. It should be noticed that $\overline{AF}$ is not the only metric that can be used. The choice is up to the network operator that might focus also on a different metric, e.g., on the EDFA with the worst failure rate acceleration performance.

7.3 Case Study

This section presents a case study where a green strategy putting EDFAs into sleep mode is applied to a specific optical backbone network under a dynamic lightpath provisioning scenario. We first explain how the green algorithm works, and then present an analysis of the various parameters impacting the average EDFA failure rate acceleration factor value

7.3.1 Green Strategy and Simulation Scenario

This simulation study is based on a green RWA strategy called Weighted Power Aware Lightpath Routing (WPA-LR) [93] tested on the COST239 optical backbone network. The provisioning scenario considers dynamic traffic where each connection request is assumed to require a full wavelength capacity.

The WPA-LR algorithm works in the following way. A separate network connectivity graph is considered for each wavelength, i.e., a wavelength plane approach is utilized. If a given wavelength is already used on a fiber link to provision a lightpath, the fiber link will not appear on that specific wavelength plane. For an incoming connection request the path at minimum cost (if any) is computed on each wavelength plane. The path that has the overall minimum cost (among the ones found on each wavelength plane) is then chosen as the route for the connection request. If no path can be found on any wavelength plane, the connection is rejected. Under the assumption that the only devices that are put into sleep mode (in order to save energy) are the EDFAs (i.e., the other network components such as transceivers, ROADMs, or higher layer electronics are considered to be always on), the cost function used in the WPA-LR algorithm works as follows. If a fiber link is not in use, its cost is equal to the power necessary to operate all the EDFAs deployed along its length (i.e., the fiber link power consumption cost). If a fiber link is in use, its routing cost becomes the product of its power consumption cost and a parameter α that varies in the range (0;1]. Values of α close to 0, encourages WPA-LR to select routes at minimum power cost, while with $0 < \alpha < 1$ WPA-LR tends to make routing choices that are a compromise between power consumption minimization and (fiber) resource efficiency maximization. When $\alpha = 1$, the WPA-LR behaves in the same way as a conventional shortest path (SP) approach, where still some energy savings can be achieved because the EDFAs that are not used can be set into sleep mode. More details about the WPA-LR strategy are available in [93].

In the simulation work on the COST239 network topology it is assumed that each fiber link comprises two unidirectional fibers each one carrying 16 wavelengths. It is assumed that wavelength conversion is not available. Connection requests are bidirectional and their source and destination pairs are uniformly chosen among the network nodes. They arrive according to a Poisson process while the service time for each connection request is exponentially distributed with an average holding time equal to 6 hours. EDFAs are placed

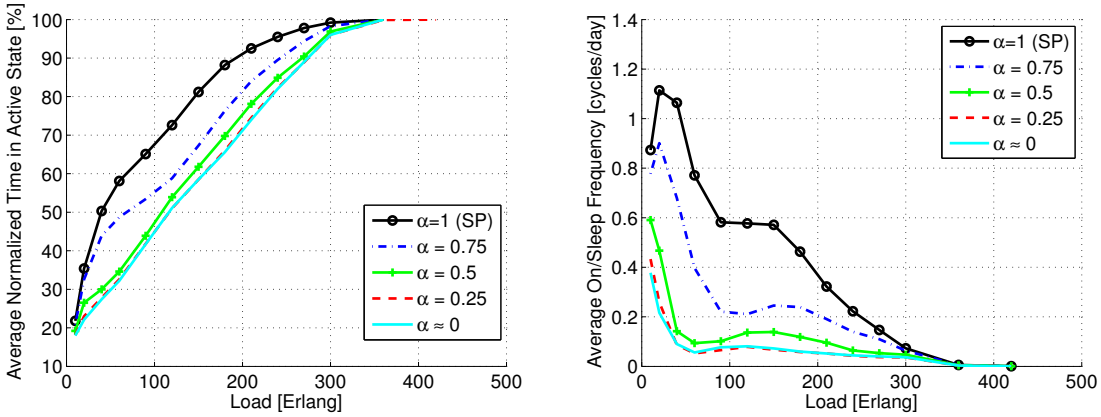


Figure 7.1: Average normalized time in percentage an EDFA is in active state (left), and average EDFA on/sleep frequency (right) as a function of the network load.

every 80 km and the power consumed by an EDFA in sleep mode is assumed to be 20% of the power when fully operational.

In the case study the traffic load is varied from 10 to 420 Erlang. These values are chosen in order to investigate different conditions where the network blocking probability does not exceed 10%. In order to measure the value of the average EDFA failure rate acceleration factor (i.e., $\overline{AF}$ as defined in equation 7.11), it is assumed that all the EDFAs deployed in the network have the same HW characteristics, while the value of the frequency and the duration of each sleep cycles are collected by simulating the WPA-LR algorithm for different values of α. Simulation results are averaged over a series of 10 experiments with 10^5 connection requests in each experiment.

7.3.2 Impact of Traffic Load on the Sleep Mode Duration and the On/Sleep Frequency

Fig. 7.1 reports the average normalized time in active state (left) and the on/sleep frequency (right) for the EDFAs in the network as a function of the load. Results are presented for different values of the parameter α. Several considerations hold in this case. First, the average normalized time in active state tends to increase with increasing load values (as expected), since more EDFAs need to be powered on in order to meet the traffic requirements. Moreover, for load higher than 350 Erlang the average normalized time in active state is nearly equal to 100%, meaning that all the EDFAs in the network are always powered on. Second, the average on/sleep frequency tends to decrease with increasing load. However, the maximum value of the on/sleep frequency does not always occur at the minimum value of load. This is due to the fact that some of the EDFAs are always in sleep mode when traffic is very low. Third, the parameter α has a strong impact on the results. In particular, when the algorithm tends to exploit short paths ($\alpha = 1$) the average normalized time in active state is higher compared to a pure power minimization approach, i.e., $\alpha = 0$, and the average on/sleep frequency is almost one order of magnitude higher since EDFAs frequently change their power state.

7.3.3 Average EDFA Failure Rate Acceleration Factor

Fig. 7.2 reports a number of level curves representing the average EDFA failure rate acceleration factor ($\overline{AF}$) (equation 7.11) as a function of the value of AF^{sleep} defined as

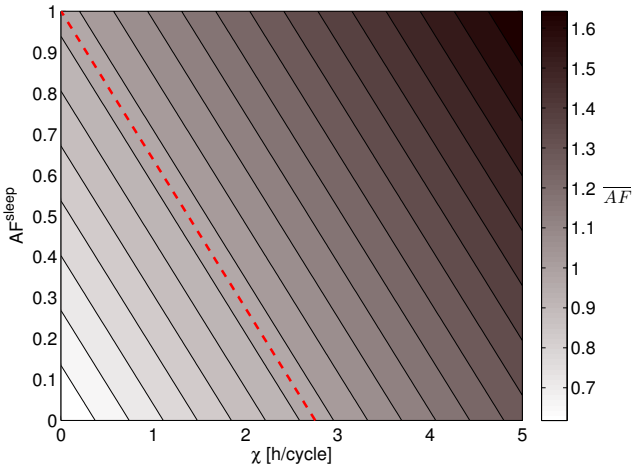


Figure 7.2: Average EDFA failure rate acceleration factor ($\overline{AF}$) as a function of AF^{sleep} and χ, with $\alpha = 0.5$ and a load of 150 Erlang.

$(\sum_{ijk} AF_{ijk}^{sleep})/I$, and χ defined as $(\sum_{ijk} \chi_{ijk})/I$. The results are obtained with $\alpha = 0.5$ at load equal to 150 Erlang.

The red dashed line highlights the level curve $\overline{AF} = 1$. The region on the left of this crossover line represents the zone where on average EDFAs in the network fail less often when compared to the case in which the WPA-LR algorithm is not used, i.e., $\overline{AF} < 1$. On the contrary, the region on the right is the zone in which $\overline{AF} > 1$, i.e., EDFAs on average are expected to fail more often when compared to the case in which the WPA-LR algorithm is not used. From the figure, it can be noticed that both AF^{sleep} and χ play a crucial role in determining the effectiveness of an energy-efficient strategy in terms of average EDFA failure rate increase/decrease. In particular, AF^{sleep} is influenced by the decrease of temperature on the device, which is expected to happen when EDFAs are in sleep mode. On the other hand χ becomes the discriminating factor, meaning that devices whose χ is very high (in this case higher than 2.7 h/cycle) will experience on average an increase of their failure rate.

Fig. 7.3, on the other hand, considers the impact of the parameter α on the position of the value for $\overline{AF} = 1$. The traffic load is equal to 150 Erlang, which corresponds to medium traffic conditions. It is interesting to notice that as the algorithm tends to target the power minimization (low values of α), the region in which $\overline{AF} < 1$ is increased. This is mainly due to the fact that, as shown in Fig. 7.1, with decreasing values of α EDFAs on average spend more time in sleep mode and the average value of their on/sleep transitions is also lower.

7.3.4 Impact of the HW parameters

The left part of Fig. 7.4 reports the different values of χ (i.e., $(\sum_{ijk} \chi_{ijk})/I$), that are required to have $\overline{AF} = 1$. The value of χ is presented as a function of the traffic load and for different values of AF^{sleep} (i.e., $(\sum_{ijk} AF_{ijk}^{sleep})/I$). The algorithm parameter α is set to 0.5.

Note that the areas below each curve represent values of χ for which $\overline{AF} < 1$. For low values of load χ is very small, i.e., $\chi < 1$ [h/cycle]. This is due to the fact that the average frequency of on/sleep transitions is relatively high as shown in Fig. 7.1. This means that

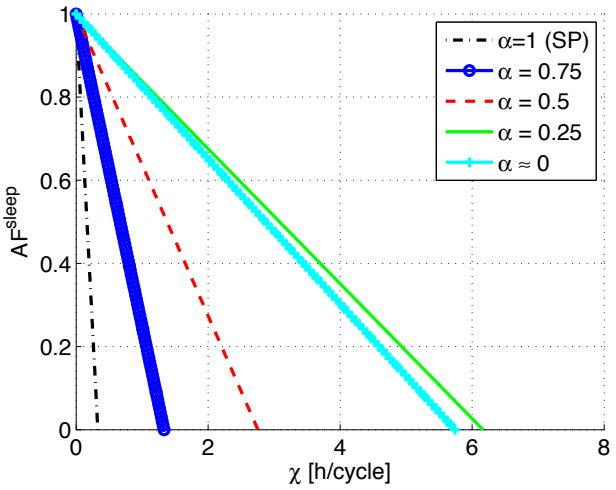


Figure 7.3: Level curves for $\overline{AF} = 1$ for different values of α with a load of 150 Erlang.

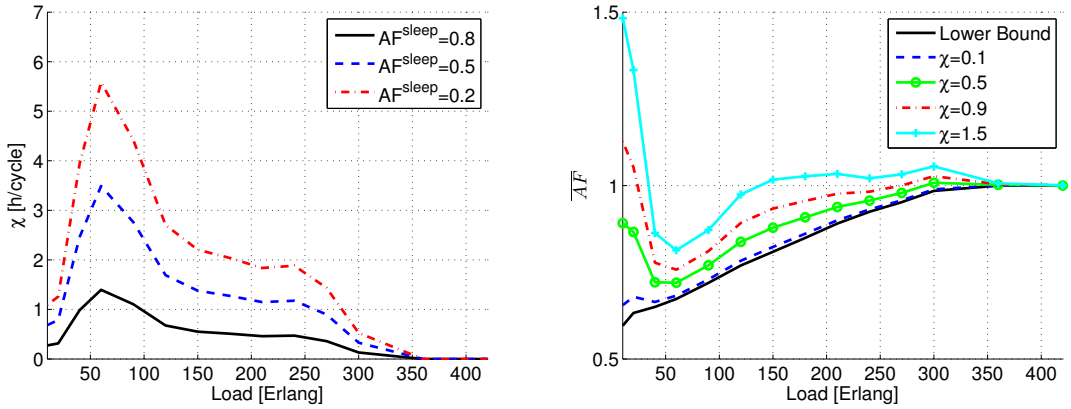


Figure 7.4: Values of χ required to have $\overline{AF} = 1$ with $\alpha = 0.5$ (left), and $\overline{AF}$ with $AF^{sleep} = 0.5$ and $\alpha = 0.5$ (right), both as a function of the load.

even if a device is equipped with better hardware capabilities to limit the thermal cycling effect, high average on/sleep frequency may still negatively impact the value of $\overline{AF}$. However, for intermediate values of load (between 20 and 125 Erlang), the frequency of cycles promptly decreases, allowing higher values for χ, i.e., up to a max of 5.5 [h/cycle], while still ensuring favorable conditions in terms of EDFAs average failure rate acceleration factor, i.e., $\overline{AF} < 1$. For high traffic values, the average normalized time EDFAs spend in active state is almost 100%, setting the value of $\overline{AF}$ to be greater than 1 even for very small values of χ. Finally, it can be observed that the trend does not change with different values of AF^{sleep} (even though the absolute values are different).

The right part of Fig. 7.4 presents values of $\overline{AF}$ for different values of the traffic load and χ when $\alpha = 0.5$ and $AF^{sleep} = 0.5$. The figure reports also a lower bound, which corresponds to the case of $\chi = 0$, i.e., no thermal cycling effect. Two considerations hold in this case. First, for increasing values of the traffic load, $\overline{AF}$ converges to one, which corresponds to the situation in which all devices are always powered on, and therefore the average EDFA failure rate acceleration factor $\overline{AF}$ remains unchanged. Second, it can be seen that when χ is increasing, the region in which the $\overline{AF} < 1$ is reduced. However, the effect is mitigated for intermediate values of load, meaning that there is a tradeoff between power efficiency and the value of the average failure rate acceleration factor.

7.4 Conclusions

The chapter considered the impact that energy-efficient techniques have on the component lifetime in an optical backbone network. The focus was on understanding the effects of putting EDFAs into sleep mode. We have developed a simple model, showing that the EDFA failure rate variations are influenced by the traffic, the energy-aware algorithm parameters, the time spent in sleep mode, the frequency of on/sleep cycles, and the characteristics of the EDFA HW. The chapter showed that frequent on/sleep transitions may have a negative impact on the EDFA failure rate. However, this work is a first step towards a more comprehensive approach, since there might be conditions under which the failure rate of a component may benefit from long (and not frequent) sleep periods. As next steps it will be worth to consider the influence of various traffic patterns (including bursty traffic), and to investigate energy-efficient algorithms targeting explicitly the increase of lifetime for optical networks.

Chapter 8

Control plane interaction for resilience scenarios

The end-to-end optical transport networks where DISCUS control plane architecture is involved moves huge volumes of data from one point to another in geographically different locations. It is not uncommon a failure in a fiber links or some component in a DISCUS node. The source of the failures is related to failure of equipment in a link (e.g., amplifiers), cuts in the fibers, e.g., because of roadworks , digging, power failure or bad weather. FCC reports publish findings that long haul networks experience annually 3 cuts for 1500 kms of fiber. Internal operator data reports even higher fiber cut failure in areas with numerous roadworks. FCC data implies a cut every four days in a typical long haul network with 45000 km of fiber. Thus, it is necessary to provide the network means of maintaining the service continuity in the presence of failures. Not only should the network be able to react to a single failure, but also to a multiple failure situation in which several link or nodes are affected simultaneously (or one after the other). In this chapter, the relation between the DISCUS control plane and network survivability is analyzed. First the control plane architecture and its main components that interact in case of failure are presented. Next, a comparative between the traditional distributed control plane approach and the centralized SDN approach is presented. Next, a use case is analyzed. Finally, a study shows which is the optimal location of a DISCUS core network controller to minimize the restoration time.

8.1 DISCUS control plane architecture

Considering the architecture of the MC node, as defined in Deliverable D6.1, it is possible to identify three main logical component for the network control plane : the access network controller, in charge of controlling the access network elements; the core network controller, in charge of controlling the elements carrying out core transmission; the network orchestrator, in charge of taking requests from the SP and translating them into high-level commands for the access and core network controllers.

The network orchestrator is defined as a parent controller or a centralized “controller of controllers”. It handles the automation of end-to-end connectivity provisioning, working at a higher, abstracted level and covering inter-domain aspects between the access and the metro/core network. Its functionalities are the following: The access network controller is in charge of controlling the access network elements, while the core controller is in charge of receiving commands from the network orchestrator and transforming them in the D-CPI for the metro/core network. For the core network controller, the technologies that are in the underlying network are Wavelength Switched Optical Network / Spectrum Switched Optical Network (WSON/SSON) which are based on the GMPLS distributed control plane.

As it can be seen in the previous definition of the entities, some feedback from the network is required for failure scenarios. The following section compares the resilience mechanisms in centralized and distributed scenarios.

8.2 Centralised vs. distributed resilience mechanism

8.2.1 Distributed control plane approach for failure scenarios

When the functions of the control plane are co-located with the nodes and do no rely on a centralized controller, they need to interact with each other in a distributed way in order to recover from network failures. Pure distributed GMPLS is an example of such distributed approach. The main concept in a distributed control plane is that the owner of a service (e.g. LSP, lambda, etc.) is the responsible of initiating any survivability actions. The main resilience mechanisms are protection and restoration. In the case of protection, when the element in control of the service detects a problem in the main path, the service is automatically switched to the backup path. Resources for both main and backup paths are always reserved, and thus, there is a huge resource consumption. The main benefit is the sub-50 ms recovery time that can be achieved with protection. The other main resilience mechanism is restoration. In this case, the nodes adjacent to the elements that have failed (in the case of a fiber cut, the switching nodes adjacent to the fiber) notify via routing protocol the failure. Then, the routing protocols keeps on disseminating the new state of the network, and all nodes become aware of the existing resource failed. Also, a second mechanism is used to speed up the recovery. The signaling mechanism (e.g. RSVP-TE in distributed GMPLS) maintains state in all the nodes of the path. When a failure happens, and the adjacent nodes become aware of the failure, using the signaling protocol, the source node (owner of the service) is notified. Then, each node that owns services that have failed recomputed a new path. Note that for the new compute path to be accurate, either the routing protocol has informed about the exact failure, or the signaling protocol carries information about the exact location of the failure. The restoration procedure takes longer that the protection, but is able to adapt to changing network circumstances, but cannot achieve an optimal resource usage after the failure, as all path computation recoveries are performed un-coordinately. Finally, restoration in a distributed control plane suffers the problem of piggybacking. As mentioned previously, when there is a failure all nodes that own a service that uses the failed element are notified. At that point, all of them compute their backup path. It may be entirely possible that the paths resulting from these almost concurrent computations share resources (e.g. a lambda in some path). This will trigger a race between signaling sessions. The fastest one gets the resource, while the following ones, need to piggyback and compute again the path. This second computation is not done immediately, but a random time is left, to avoid concurring again in a computation. The problem of piggybacking can lead to high restoration times when the network is highly utilized. A centralized approach for the computation can solve the problems of the distributed control plane.

In spite of the problems of the recovery of a distributed control plane, it is a very robust scheme, as it can survive many failure scenarios (including controllers).

Figure 8.1 shows the message exchange between nodes in the control plane. Let's assume there is a path from N_0 to N_3. When there is a failure at a node, for instance N_2, this node sends an RSVP Error message to the head-end node (in this case N_0). Once the head-end node receives the failure notification, it computes a new path avoiding this failed node and establishes a new connection. This is done using the RSVP Path-Resv message exchange, as shown in Figure 8.1 .

The restoration in a distributed control plane suffers the problem of piggybacking. As mentioned previously, when there is a failure all nodes that own a service that uses the failed

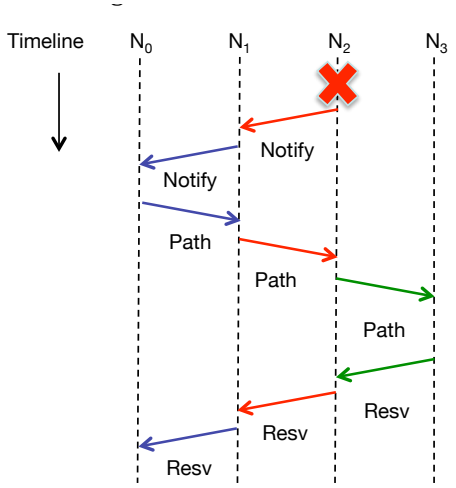


Figure 8.1: Control plane restoration process.

element are notified. At that point, all of them compute their backup path. It may be entirely possible that the paths resulting from these almost concurrent computations share resources (e.g. a lambda in some path). This will trigger a race between signaling sessions. The fastest one gets the resource, while the following ones, need to piggyback and compute again the path. This second computation is not done immediately, but a random time is left, to avoid concurring again in a computation. The problem of piggybacking can lead to high restoration times when the network is highly utilized. A centralized approach for the computation can solve the problems of the distribution control plane. However, there is a single point of failure in the central control plane entity. In spite of the problems of the recovery of a distributed control plane, it is a very robust scheme, as it can survive many failure scenarios (including controllers).

8.2.2 Centralized SDN approach for failure scenarios

The centralized SDN control can also perform protection and restoration. Let's assume three different schemes: (1) data plane protection, (2) controller based protection and (3) controller based restoration. Figure 8.2 shows the approaches.

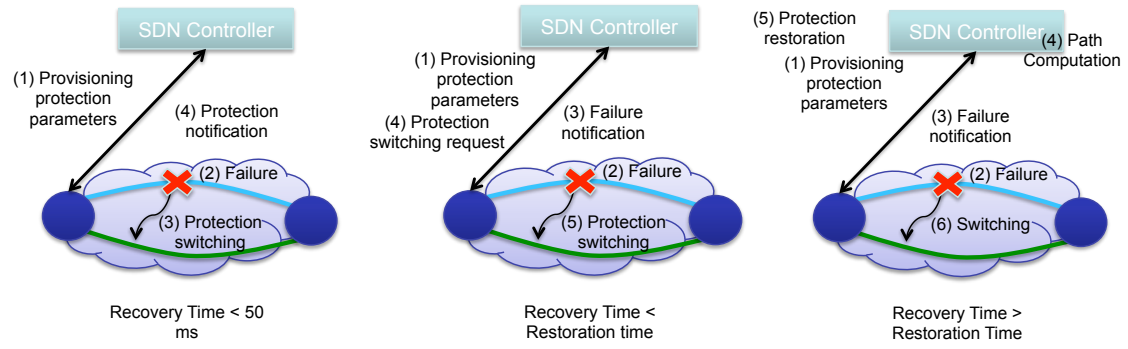


Figure 8.2: Protection schemes in SDN environments

The data plane protection mechanism works similar to Network Management System protection, but driven from the SDN controller. The SDN controller computes and configures

a working and a protection path, which are instantiated in the network elements. When there is a failure, the network elements can notify the controller, which can configure the protection path to recover from the failure in the network. The protection, as well with the distributed control plane, can be performed by the data plane in case sub-50 ms recovery time is needed. In this case, the control plane is in charge only of computing and setting up the working and protection paths, but it is not involved at failure time.

The controller, according to ITU-T [53], can be used to perform a controller-based protection, or controller-based restoration.

In the controller based protection, SDN controller is in charge of provisioning both working and protection path and instructs the data plane to notify the controller whenever a failure happen. When the failure happens, the SDN controller commands to switch to the protection path. This mechanism is not as fast as the data plane protection, but faster than a restoration as it does not need to compute path. The advantage is that it lowers the data plane requirements.

In the controller based restoration, the SDN controller is notified of all the failures. The main difference with the distributed case is that the SDN controller is the owner of all the services. When the SDN controller receives the notification of the failed element, it will look in its database and see how many services are affected by the failure. Then, it performs the computation of new paths for all the services. Such computation can be performed in a single computational step, which allows to apply optimization techniques. The result of the recovery of multiple paths via SDN controller can lead to a very high restorability degree, and an optimal use of resources. Furthermore, as the controller centralizes the provisioning and computation, no piggybacking is needed. All paths computed by the SDN controller will be valid.

The main problem of the centralized SDN controller approach is its high dependability on the controller, and having a path to the controller. For the availability of the controller, dual-controller approach (service-backup) is recommended for deployments.

8.3 Impact of a centralized or distributed control plane in DISCUS

DISCUS defined use cases in D6.1. One of this scenarios is the PON protection with a dual-homed N:1 OLT scheme. This scenario emulates the provisioning of a dedicated assured L1 service over PON, describing the interaction between ONU, OLT, node controller, and network control plane. We assume this service is a point-to-point service between any two points in the network. In order to understand the requirements on all interfaces involved we carry out an analysis of the upstream and downstream control messages required. The interactions between the control plane elements is the following:

1. The fibre cut is identified by a fault management system that triggers an alarm.
2. The access controller in MC node 1 informs the network orchestrator about the failure.
3. The orchestrator informs the core controller to setup the pre-calculated core backup paths corresponding to the failure.

4. The orchestrator then communicates to the access controller in the MC node hosting the standby OLT, passing the information required to activate the OLT as well as the appropriate protection paths that need to be activated in the node.
5. In the meantime the core controller will configure the core nodes
6. The access controller will configure the access elements
7. The core controller will then provide feedback to the orchestrator.
8. The access controller of the standby node on the access protection.

Figure 8.3 presents graphically the steps just described in the reference architecture.

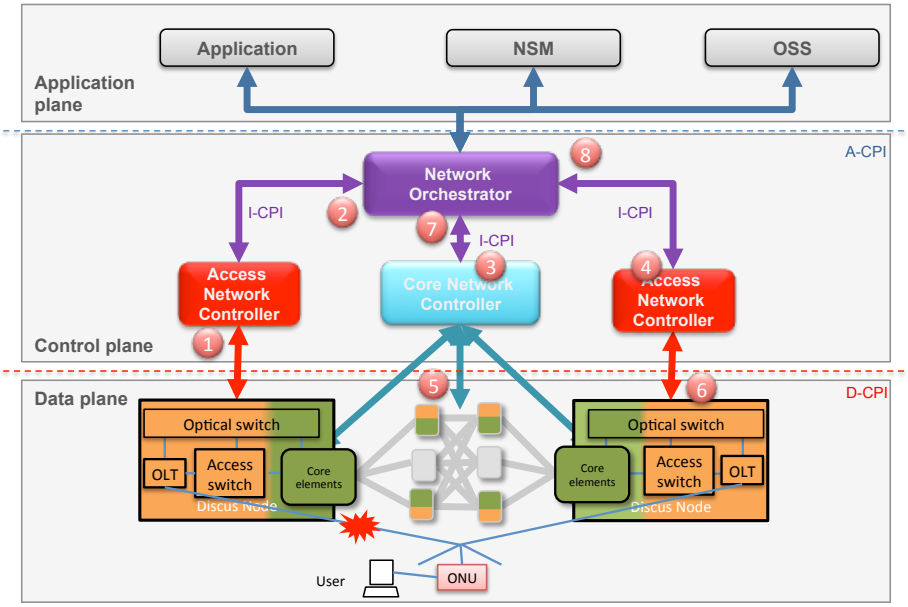


Figure 8.3: PON protection with a dual-homed N:1 OLT scheme.

Two options were considered in the definition of the control plane solution for the core segment (distributed or centralized control plane). However, in the definition of the control plane for the access-metro region, DISCUS control plane is configured from the access network controller. This controller is local and all elements that it is managing are co-located with it. All actions that happen in the local area are not impacted by the definition of a distributed or central control plane in the core part. For instance, when there is a fibre cut, this is identified by a fault management system that triggers an alarm. This information is local to the access controller in MC node 1, which informs the network orchestrator about the failure. All actions that are defined in the network orchestrator or the access controller are not impacted.

On the other hand all actions that has to cross between multiple optical islands are impacted by the utilization of the control plane.

8.4 Performance analysis of DISCUS control plane

8.4.1 Distributed control plane delay analysis

The control plane performance is analyzed considering a worst-case scenario when a failure happens. Let us assume a network where the longest shortest path contains Nmax hops. The worst-case scenario for a failure is when the error is in the last hop of the path. Figure 8.4 shows the time consumed for this scenario.

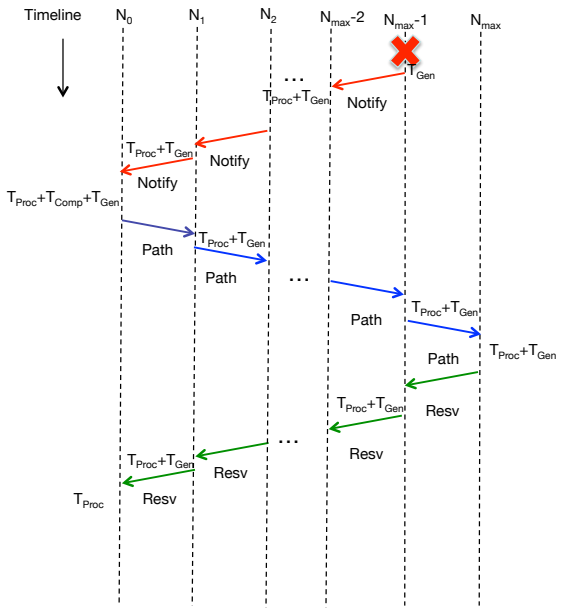


Figure 8.4: Distribute control plane delay analysis for the worst case scenario.

After a failure happens at N_{max}-1, this node notifies to the head-end node the failure, so the path can be restored. This means that N_{max}-1 Notify messages are generated. Each Notify message is generated and processed. Therefore, the control plane delay to alert the head-end node of the failure is (N_{max}-1) (T_{gen}+T_{proc}+T_{prop}), where, T_{gen} is the generation time for the Notify message, T_{proc} is its processing time, and T_{prop} is the propagation delay of the packet. We have assumed the same time to process and generate all RSVP messages as done in previous work [95]. Once the message is received at the head-end, the end-to-end path must be computed if there is no back-up path precomputed. This time is defined as T_{comp}. A similar reasoning can be done for the Path and Resv messages, so the provisioning time is 2*N_{max} (T_{gen}+T_{proc}+T_{prop}). Therefore, the recovery time for the worst-case scenario is:

$$T_{rec-wc} = (3N_{max} - 1)(T_{gen} + T_{proc} + T_{prop}) + T_{comp} \quad (8.1)$$

8.4.2 Centralized control plane delay analysis

The centralized control plane assumes that there is an SDN controller, which can configure the network elements. This entity could follow any of the previous schemes, but for the sake of comparison let us assume a controller based restoration scheme. As previously

stated, if the controller instantiates the back-up path too, the recovery time is like a protection in the distributed control plane scenario. The following figure presents the controller based restoration scheme scenario:

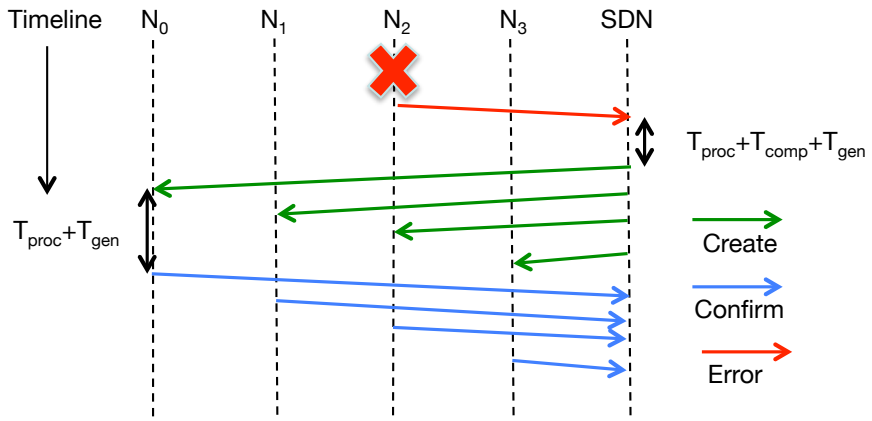


Figure 8.5: Centralized control plane delay analysis.

The node that detects the failure sends an Error message to the SDN controller to notify the failure event. Afterwards, the SDN controller processes the message and it computes the restoration path for the failed path (in case it was not pre-computed). The SDN controller sends a Create message to each node. Once the node processes the message, it confirms the resource allocation to the SDN controller. Based on the workflow just described, we can state that the recovery time for this scenario is the following:

$$T_{rec-sdn} = 3T_{prop} + 2(T_{gen} + T_{proc}) + T_{comp} \tag{8.2}$$

Based on the previous analysis, it is clear that the location of the SDN controller is important to minimize the delay of the transmission to and from the SDN controller. The aim of the analysis that is presented the remaining of this section is to find the central point of a reference network to define which is the best location to position the SDN controller.

The reference network of Telefónica I+D is shown in Fig. 8.6. A total number of 33 nodes are present out of which the longest link is the one between node 6 and 32 with a total length of 1900 km. There are 106 links and the average nodal degree is 3.2121.

The network in Fig. 8.6 has been analyzed on the node-to-node distance and the number of hops between nodes. Both have been determined using the $graphshortestpath-function$ in MATLAB that uses Dijkstra as default algorithm.

The results on the node-to-node distance are shown in Fig. 8.7. The x-axis shows the shortest distance determined by Dijkstra from a particular node to each of the 32 other nodes in the network. The y-axis then shows the percentage of the total amount of nodes such that it may be easily determined how many nodes are within a particular distance from a particular node. The main outcome of Fig. 8.7 is that it indicates node 27 as having the shortest distance to 90% of the nodes in TID's reference network. The second closest node in this case is node 28, and the third one is node 12. The maximum amount of hops in the TID reference network is 5 from one node to another one which can be achieved from nodes 12, 17, 25, 26, and 27. The ideal location of the SDN controller can now only be either node 12 or node 27 which is determined by the lowest total amount of hops as shown in Fig. 8.8. This figure shows the amount of hops (nodes) that have to be made in

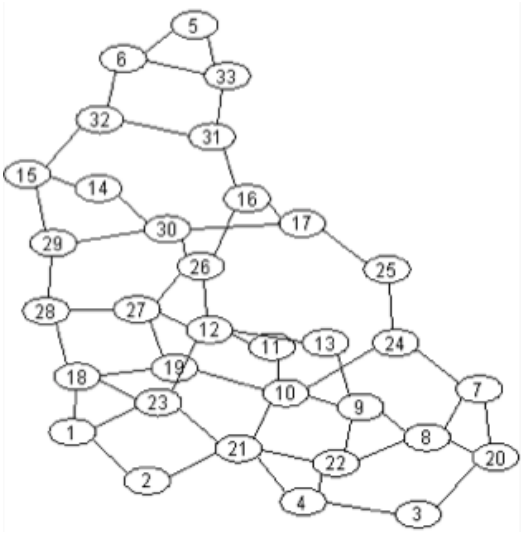


Figure 8.6: The reference network of Telefónica I+D for SDN placement.

order to reach a particular node. It is clearly shown in Fig. 8.8 that node 12 (with a total of 88 hops; node 27 has 96 hops) is the ideal location of the SDN controller in TID's reference network.

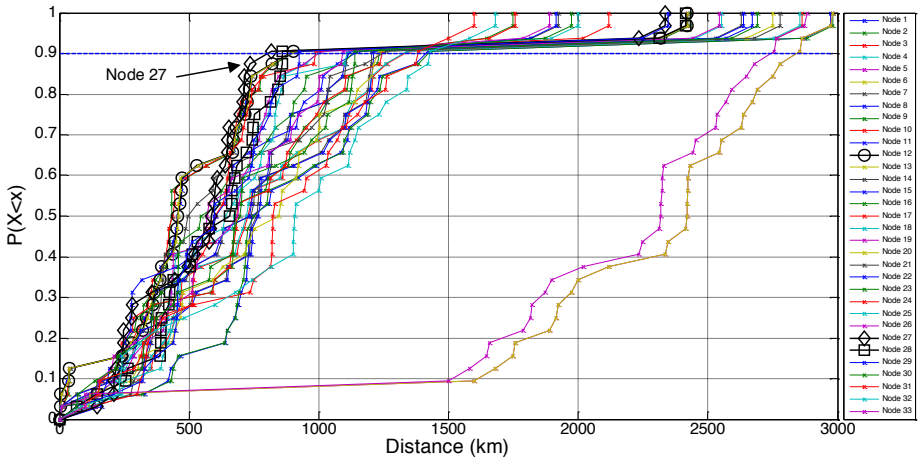


Figure 8.7: Distance between nodes vs. Percentage of nodes in TID's reference network.

8.5 Comparison between both approaches

Once both scenarios are analysed this section compares both control plane approaches based on a given scenario.

8.5.1 GMPLS control plane delay values

The value of T_{proc} is set to 50 ms and the one of T_{gen} to 10 ms [95] . Moreover, the propagation delay as well as the queuing delays are assumed to be negligible 3. We have compared the values in the reference with the results in Telefonica GMPLS control

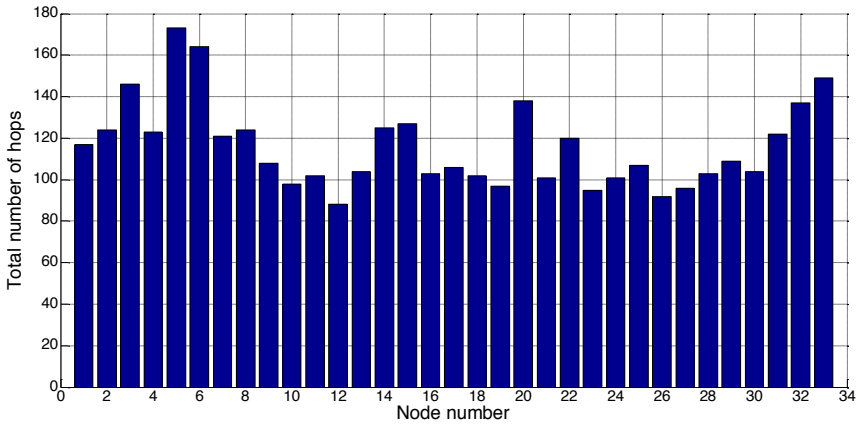


Figure 8.8: Total number of hops in TID's reference network (shown per node).

plane. The Telefonica I+D control plane test bed is composed by GMPLS nodes with software developed internally. The experimental setup is built with emulated nodes, which run in an Ubuntu server Linux distribution. Each emulated node implements a GMPLS stack (including RSVP, OSPFv2, and PCEP) and a Flexible Node emulator. Each GMPLS controller is a virtual machine and all are running is a server with two processor Intel Xeon E5-2630 2.30GHz, 6 cores each, and 192 GB RAM. Each VM has 1GB RAM. The GMPLS control plane stack and Flexible Node emulator are being developed in Java 1.6. Note that there is no hardware associated to this domain, only an emulation of the nodes. When a path is provision on a six-hop path, the RSVP message exchange in an intermediate hop is shown in the next figure:

68	29.817481	192.168.1.1	192.168.1.3	RSVP	214 PATH Message.
71	29.864739	192.168.1.3	192.168.1.9	RSVP	214 PATH Message.
75	30.117845	192.168.1.9	192.168.1.3	RSVP	254 RESV Message.
76	30.175678	192.168.1.3	192.168.1.1	RSVP	310 RESV Message.

Figure 8.9: Control plane emulated scenario.

The delay between two consecutive PATH messages is 47,3 ms and RESV message is 57,8 ms. These values are in the order of the 50 ms assumed for this analysis.

8.5.2 Computation time delay

Once the processing time is assessed with literature and a validation in the lab, we have analysed which is the computation time for a state of the art algorithm using a low cost server. The aim is to have a value for the computation time without requiring to deploy a datacentre. The IA-RWA algorithm used is based on K shortest path (KSP) for the routing, First-Fit for the wavelength allocation and for the impairment validation we've used the same method as in [44]. This method is valid for 10Gbps connections. The server where the tests are done is a HP Proliant 320, Dual Core Intel(R) Xeon(R) - 2.66GHz and 4GB RAM. The RWA module is implemented in C++ against a MySQL database. The server and the database are in the same server.

The KSP algorithm is a variation of Yen's version. The metric for the KSP is the number of hops. The physical impairment validation can be done based on the number of spans. There are two scenarios where the implementation has been tested without load and with

load. In the first experiment 81 requests are sent between the node 7 and node 3 in the Telefonica reference network. The average time is 23ms. Figure 8.10 shows the histogram with the results, while Figure 8.11 shows the computation time with a high load in the network.

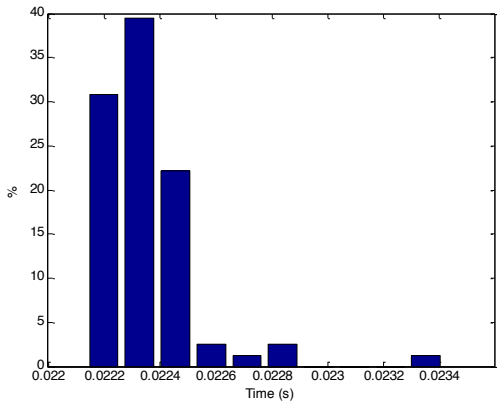


Figure 8.10: Histogram with the computation time in a scenario without load.

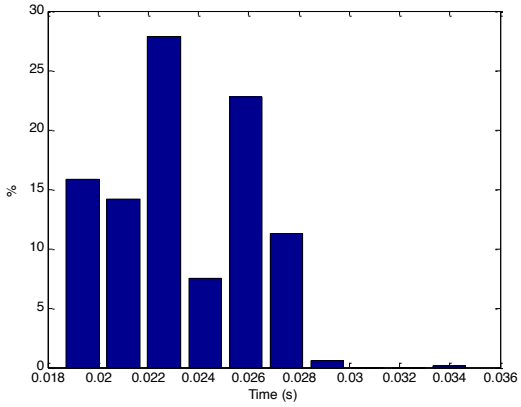


Figure 8.11: Histogram with the computation time in high load scenario.

The average time for the request is 23 ms with a variance time of 6.9071 μs. The relative error for a confidence interval of α=95%, is 0.816 μs. We can see that the results are stable. Figure 8.12 shows the network occupation.

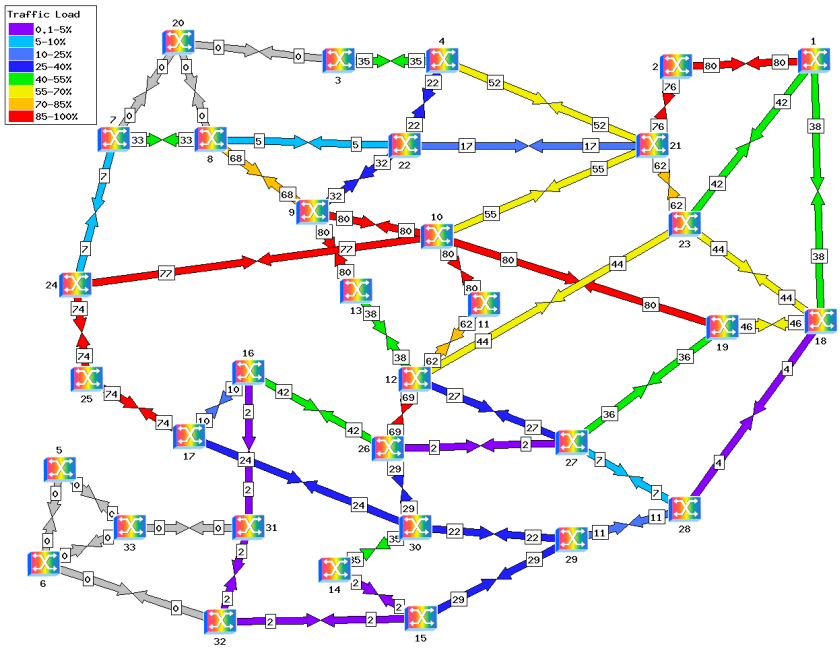


Figure 8.12: Number of wavelength with the load.

As it can be seen in the previous figure, there are some links that are overloaded, but this does not impact the algorithm performance. This is because the main part of the computation time is due to the KSP algorithm. The First-Fit can take more time, but the

main contribution is KSP.

8.5.3 Assessment of central and distributed solution

Based on the values previously defined and justified, the recovery delay for the centralized and the distributed recovery are compared. Figure 8.13 shows the recovery time for both approaches.

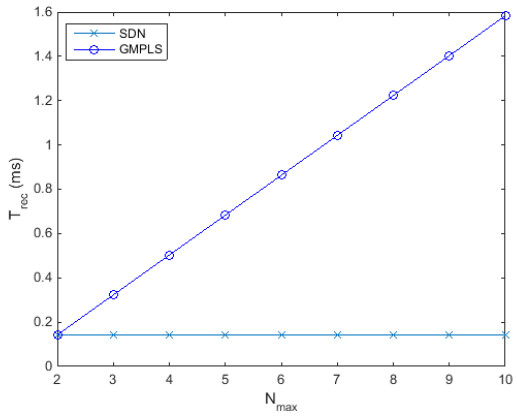


Figure 8.13: Recovery time for SDN and GMPLS without propagation delay.

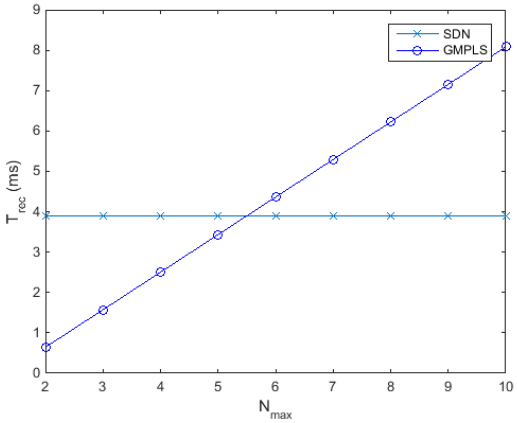


Figure 8.14: Recovery time for SDN and GMPLS with propagation delay.

GMPLS is as fast as the SDN controller only when the number of hops is 2. The reason is that the SDN controller is at two hops from any failure. The reason is that the failure is notified to the controller, which, afterwards, configures the nodes with the failure. We assumed that the propagation delay in the control plane is negligible. However, if this value were 250ms and the average number of hops to the SDN controller where 5, the delay figure would be different. Figure 8.14 shows the recovery time.

The SDN approach could minimize the computation time. However for current analysis the impact of the algorithm is so small that it won't benefit this scenario.

8.6 Conclusions

Control plane architecture defined in DISCUS project considered distributed control plane for the core part. The distributed control plane is more resilient that the centralized approach, but if the SDN controller is in two datacentres with multiple VMs, the resilience problems are minimized. Based on the analysis done in this document with realistic scenarios, the controller-based approach is very promising as it minimized the recovery time. This validates the DISCUS control plane architecture not only in terms of functional features, but also in resilience scenarios.

Chapter 9

Conclusions

This deliverable presented a number of optimization models and methods that can be used to dimension the core network segment of the DISCUS architecture. The focus of these models has been mainly on cost minimisation and on the maximisation of the resiliency levels that can be offered to the provisioned services. The scenarios considered included both network design and in-operation optimization problems.

The contribution of the deliverable is manyfold. This was done on purpose since the main objective of the set of studies carried out so far in the project was to assess the performance of a wide set of options in terms of core network optimization strategies. The plan is to identify the most promising ones to be used in the optimization work for the final consolidated/integrated optical core design phase, that will be presented in Deliverable D7.7. A few concluding remarks about the various contributions presented in this deliverable are presented next.

Chapter 2 provided a consolidated model of cost and hardware including core photonic switching, signal regeneration, and Raman amplification, while Chapter 3 presented methods that can be used to solve multi-layer core network design problems in order to assess the impact of the DISCUS optical island concept. It was found that optical islands outperform architectures based on aggregating (grooming) traffic towards an inner core once the traffic volume exceeds a certain threshold. These results are dependent on the cost model, the number of metro-core nodes, and the available channel capacities (40G, 100G, 400G).

Chapter 4 presented planning approaches for providing core-network survivability in the presence of single link- and node- failures, individual node component faults and deliberate physical-layer attacks targeting service disruption. On the other hand Chapter 5 looked more into the problem of providing resiliency to services in the presence of dynamic traffic where both single and double link failures scenarios where considered.

Chapter 6 presented a study on the dimensioning of metro/core networks with dual-homed access under the assumption of single metro/core node failure scenario. The results demonstrate that multilayer restoration combined with the right amount of transponder overprovisioning allows to achieve average connection availability levels very similar to the ones achieved by dedicated path protection but with lower service blocking levels and at the same time using a lower number of WDM transponders.

Chapter 7 evaluated the impact that energy-efficient techniques have on the component lifetime in an optical backbone network. The focus was on understanding the effects of putting EDFAs into sleep mode. The study showed that frequent on/sleep transitions may have a negative impact on the EDFA failure rate. Finally, Chapter 8 presented a study evaluating centralized versus distributed control plane architectures in terms of recovery time offered to services. It was found that the propagation delay of control messages has a non trascurable impact on the overall recovery time. For this reason the controller placement within the network needs to be carefully considered.

Appendix A

Acronyms

Acronym	Explanation
AA-DPP	Attack-Aware Dedicated Path Protection
AA-DPP-H	Attack-Aware Dedicated Path Protection Heuristic
AF	Accelleration Factor
AG	Attack Group
AkNLPP	All-Pairs k Node-Disjoint Length-Bounded Paths Problem
AoD	Architecture on Demand
AWG	Arrayed Waveguide Grating
BPSK	Binary Phase-Shift Keying
BV-WSS	Bandwidth-Variable Wavelength Selective Switches
BR	Backup Reprovisioning
BSPP	Backup Server via Physically disjoint Path
CO	Central Office
CTE	Coefficient of Temperature Expansion
DC	Data Center
DGE	Digital Gain Equalizer
DH-DPP	Dual Homin Dedicated Path Protection
DH-PR-AVG	Dual Homing Path Restoration Average
DH-PR-MAX	Dual Homing Path Restoration Max
DH-PR-UNP	Dual Homing Path Restoration Unprotected
DP-QPSK	Dual Polarization Quadrature Phase-Shift Keying
DPP	Dedicated Path Protection
DPP-EFS	Dedicated Path Protection with Enforced Fiber Switching
DPP-SP	Dedicated Path Protection with Shortest Path routing
DWDM	Dense WDM
EDFA	Erbium Doped Fiber Amplifiers
EFS	Enforced Fiber Switching
FCC	Federal Communication Commission
FIT	Failure In Time
FS	Fiber Switching
GB	Gigabyte
Gbps	Gigabit per second
HW	Hardware
H-W	Hard-Wired
KSP	K shortest path
IA-RWA	Impairment Aware Routing and Wavelength Assignment
ICU	Idealist Cost Unit
IP	Internet Protocol
ILP	Integer Linear Program
IX	Internet Exchange, Peering Point

LE	Local Exchange
LP	Linear Program
LR-PON	Long Reach PON
LSP	Labeled Switched Path
MC	Metro-core
Mbps	Megabit per second
MDD	Minimum Distance Design
MDT	Mean Down Time
MEMS	Micro Electro Mechanical Switches
MIP	Mixed Integer Program (MILP)
MILP	Mixed ILP
GMPLS	Generalized Multi-Protocol Label Switching
MPLS	Multi-Protocol Label Switching
MPLS-TP	MPLS Transport Profile
ODN	Optical Distribution Network
OEO	Optical-to-Electrical-to-Optical
OLT	Optical Line Termination
OLA	Optical Line Amplifier
ONU	Optical Network Unit
OOK	On-Off Keying
OSPF	Open Shortest Path First
OXC	Optical Cross Connect
PCE	Path Computation Element
PCEP	Path Computation Element communication Protocol
PON	Passive Optical Network
PR	Path Restoration
QPSK	Quadrature Phase-Shift Keying
ROADM	Reconfigurable Optical Add Drop Multiplexer
RSVP	Resource Reservation Protocol
RWA	Routing and Wavelength Assignment
SDN	Software Defined Networks
SD	Standard Definition
SSS	Spectrum Selective Switch
SSON	Spectrum Switched Optical Networks
Tbps	Terabit per second
TE	Traffic Engineering
UK	United Kingdom
VOA	Variale Optical Attenuator
VTD	Virtual Topology Design
WDM	Wavelength Division Multiplexing
WP2/WP4/WP7	DISCUS Work Package 2/4/7
WPA-LR	Weighted Power Aware Lightpath Routing
WSON	Wavelength Switched Optical Networks
WSS	Wavelength Selective Switch

Appendix B

Versions

Version[1]	Date submitted	Comments
V1.0	30/04/2015	First version sent to the commission

[1] Last row represents the current document version

Bibliography

[1] Gurobi Optimizer. URL http://www.gurobi.com/.

[2] Net2Plan – The open-source network planner. URL http://www.net2plan.com.

[3] IDEALIST, Deliverable 1.1, Elastic Optical Network Architecture: reference scenario, cost and planning. Technical report, IST IP IDEALIST: Industry-Driven Elastic and Adaptive Lambda Infrastructure for Service and Transport Network, 2013.

[4] DISCUS, Deliverable 7.2, Optical Island Description. Technical report, The DISCUS Project (FP7 Grant 318137), 2013.

[5] DISCUS, Deliverable 2.4, Progress report on traffic and service modelling. Technical report, The DISCUS Project (FP7 Grant 318137), 2014.

[6] DISCUS, Deliverable 2.6, Architectural optimization for different geo-types . Technical report, The DISCUS Project (FP7 Grant 318137), 2014.

[7] DISCUS, Deliverable 6.1, First Report on the Specification of the Metro/Core Node Architecture. Technical report, The DISCUS Project (FP7 Grant 318137), 2014.

[8] DISCUS, Deliverable 6.2, Report on the design and control mechanism of a 3-stage optical switch. Technical report, The DISCUS Project (FP7 Grant 318137), 2014.

[9] DISCUS, Deliverable 7.1, Power Consumption Studies. Technical report, The DISCUS Project (FP7 Grant 318137), 2014.

[10] DISCUS, Deliverable 7.2, Preliminary quantitative results for flat optical network. Technical report, The DISCUS Project (FP7 Grant 318137), 2014.

[11] DISCUS, Deliverable 2.8, DISCUS end-to-end techno-economic model. Technical report, The DISCUS Project (FP7 Grant 318137), 2015, in preparation.

[12] DISCUS, Deliverable 6.5, Final report on the specification of the metro/core node architecture. Technical report, The DISCUS Project (FP7 Grant 318137), 2015, in preparation.

[13] A. Farrel, et al. A Path Computation Element (PCE) Based Architecture, 2006. URL https://tools.ietf.org/html/rfc4655.

[14] Sandu Abeywickrama, Marija Furdek, Paolo Monti, Avishek Nag, and Elaine Wong. Dual-homing based protection for enhanced network availability and resource efficiency. In *Asia Communications and Photonics Conference 2014*, page ATh1H.6. Optical Society of America, 2014.

[15] J. Ahmed and S. Nawaz. Proc. ncet. In *POSE: A New Discrete Event Optical Simulator for the Java Platform*, 2004.

[16] J. Ahmed, C. Cavdar, P. Monti, and L. Wosinska. A dynamic bulk provisioning framework for concurrent optimization in pce-based wdm networks. *Lightwave Technology, Journal of*, 30(14):2229–2239, July 2012.

[17] J. Ahmed, C. Cavdar, P. Monti, and L. Wosinska. Hybrid survivability schemes achieving high connection availability with a reduced amount of backup resources [invited]. *Optical Communications and Networking, IEEE/OSA Journal of*, 5(10):A152–A161, Oct 2013.

[18] Jawwad Ahmed, Paolo Monti, and Lena Wosinska. Benefits of connection request bundling in a pce-based wdm network. In *Proc. of European Conference on Networks and Optical Communications (NOC)*, 2009.

[19] Jawwad Ahmed, Paolo Monti, Lena Wosinska, and Salvatore Spadaro. Enhancing restoration performance using service relocation in pce-based resilient optical clouds. In *Optical Fiber Communication Conference*, page Th3B.3. Optical Society of America, 2014.

[20] Norberto Amaya, Georgios Zervas, and Dimitra Simeonidou. Architecture on demand for transparent optical networks. In *ICTON*, pages 1–4, 2011.

[21] Norberto Amaya, Georgios Zervas, and Dimitra Simeonidou. Introducing Node Architecture Flexibility for Elastic Optical Networks. *IEEE/OSA Journal of Optical Communications and Networking*, 5(6):593–608, 2013.

[22] R. Andersen, Fan Chung, A. Sen, and Guoliang Xue. On disjoint path pairs with wavelength continuity constraint in wdm networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, pages –535, March 2004.

[23] Takashi Anzawa, Qiang Yu, Masanori Yamagiwa, Tadahiro Shibutani, and Masaki Shiratori. Power cycle fatigue reliability evaluation for power device using coupled electrical-thermal-mechanical analysis. In *Proc. of the ITHERM, Orlando, USA*, May 2008.

[24] S. Arrhenius. *Über die Reaktionsgeschwindigkeit bei der Inversion von Rohrzucker durch Säuren*. Wilhelm Engelmann, 1889.

[25] F. Bayle and A Mettas. Temperature acceleration models in reliability predictions: Justification & improvements. In *Proc. of the RAMS, San Jose, USA*, January 2010.

[26] Pietro Belotti, Antonio Capone, Giuliana Carello, and Frederico Malucelli. Multi-layer MPLS network design: The impact of statistical multiplexing. *Computer Networks*, 52 (6):1291–1307, 2008.

[27] Richard Blish and Noel Durrant. Semiconductor device reliability failure models. *International Sematech Technology Transfer #00053955A-XFR*, May 2000.

[28] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti. Energy efficiency in the future internet: a survey of existing approaches and trends in energy-aware fixed network infrastructures. *IEEE Communications Surveys & Tutorials*, 13(2):223–244, 2011.

[29] L. Chiaraviglio and M. Listanti. On the interplay between sleep modes and device lifetime in telecommunication networks. In *Proc. of the EINS Workshop on Understanding the Interplay Between Sustainability, Resilience, and Robustness in Networks (USRR), Ghent, Belgium*, April 2014.

[30] Luca Chiaraviglio, Antonio Cianfrani, Angelo Coiro, Marco Listanti, Josip Lorincz, and Marco Polverini. Increasing device lifetime in backbone networks with sleep modes. In *Proc. of the SoftCOM, Primosten, Croatia*, September 2013.

[31] Biing-Feng Wang Chih-Chiang Yu, Chien-Hsin Lin. Improved algorithms for finding length-bounded two vertex-disjoint paths in a planar graph and minmax k vertex-disjoint paths in a directed acyclic graph. *Journal of Computer and System Sciences*, 76:697–708, 2010.

[32] X. Chu, B. Li, and Z. Zhang. A dynamic rwa algorithm in a wavelength-routed all-optical network with wavelength converters. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1795–1804 vol.3, March 2003.

[33] Cisco. Cisco Visual Networking Index: Forecast and Methodology, 2013 – 2018, 2015. URL `http://http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.html`.

[34] (L. F.) Coffin Jr. and U.S. Atomic Energy Commission and General Electric Company. *A Study of the Effects of Cyclic Thermal Stresses on a Ductile Metal*. Knolls Atomic Power Laboratory, 1953.

[35] Ayse K Coskun, Richard Strong, Dean M Tullsen, and Tajana Simunic Rosing. Evaluating the impact of job scheduling and power management on processor lifetime for chip multiprocessors. In *Proc. of the SIGMETRICS/Performance, Seattle, USA*, June 2009.

[36] C. Develder, M. De Leenheer, B. Dhoedt, M. Pickavet, D. Colle, F. De Turck, and P. Demeester. Optical networks for grid and cloud computing applications. *Proceedings of the IEEE*, 100(5):1149–1167, May 2012.

[37] C. Develder, J. Buysse, B. Dhoedt, and B. Jaumard. Joint dimensioning of server and network infrastructure for resilient optical grids/clouds. *Networking, IEEE/ACM Transactions on*, 22(5):1591–1606, Oct 2014.

[38] M. Nishan Dharmaweera, Rajendran Parthiban, and Y. Ahmet Şekercioğlu. Towards a power-efficient backbone network: The state of research. *IEEE Communications Surveys & Tutorials*, 2014. accepted for publication.

[39] E.W. Dijkstra. A Note on Two Problems in Connection with Graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

[40] Matija Džanko, Marija Furdek, Norberto Amaya, Georgios Zervas, Branko Mikac, and Dimitra Simeonidou. Self-healing Optical Networks with Architecture on Demand Nodes. In *European Conference and Exibition on Optical Communication*, pages 1–3, 2013.

[41] Matija Džanko, Marija Furdek, Georgios Zervas, and Dimitra Simeonidou. Evaluating Availability of Optical Networks Based on Self-Healing Network Function Programmable ROADMs. *IEEE/OSA Journal of Optical Communications and Networking*, 6(11):974–987, 2014.

[42] M. Dzanko, M. Furdek, N.A. Gonzalez, G. Zervas, B. Mikac, and D. Simeonidou. Experimental demonstration and benefits of self-healing hard-wired and synthetic roadms. In *Optical Fiber Communications Conference and Exhibition (OFC), 2014*, pages 1–3, March 2014.

[43] W. Engelmeier. *Solder Joints in Electronics: Design for Reliability*. URL `http://analysistech.com/downloads/SolderJointDesignForReliability.pdf`.

[44] D. Alvarez et al. Utilization of temporary reservation of path computed resources for multi-domain pce protocols in wdm networks. In *Network of the Future Conference*, Nov 2011.

[45] M. Garrich et al. Architecture on demand: synthesis and scalability. In *Proc. of IEEE Optical Network Design and Modeling*, ONDM, 2012.

[46] N. Amaya et al. Architecture on demand for transparent optical networks. In *Proc. of IEEE International Conference on Transparent Optical Networks*, ICTON, 2011.

[47] Y. Wang et al. Multi-granular optical switching: A classified overview for the past and future. *IEEE Communications Surveys and Tutorials*, 14(3):698–713, 2013.

[48] Mabel P. Fok, Z. Wang, Y. Deng, and Paul R. Prucnal. Optical layer security in fiber-optic networks. *IEEE T. Inf. Foren. Sec*, 6(3):725–736, 2011.

[49] Marija Furdek, Nina Skorin-Kapov, Szilard Zsigmond, and Lena Wosinska. Vulnerabilities and security issues in optical networks. In *ICTON*, pages 1–4, 2014.

[50] Miquel Garrich, Norberto Amaya, Georgios Zervas, Paolo Giaccone, and Dimitra Simeonidou. Architecture on Demand: Synthesis and Scalability. In *International Conference on Optical Network Design and Modeling*, pages 1–6, 2012.

[51] Miquel Garrich, Norberto Amaya, Georgios Zervas, Paolo Giaccone, and Dimitra Simeonidou. Power Consumption Analysis of Architecture on Demand. In *European Conference and Exhibition on Optical Communication*, pages 1–3, 2012.

[52] Filip Idzikowski, Luca Chiaraviglio, Raúl Duque, Felipe Jimenez, and Esther Le Rouzic. Green horizon: Looking at backbone networks in 2020 from the perspective of network operators. In *Proc. of the ICC, Budapest, Hungary*, June 2013.

[53] ITU-T meeting Bundang, South Korea. Controller based protection, 2015.

[54] J.-L. Izquierdo Zaragoza and P. Pavon-Marino. Assessing ip vs optical restoration using the open-source net2plan tool. In *Telecommunications Network Strategy and Planning Symposium (Networks), 2014 16th International*, pages 1–6, Sept 2014.

[55] JEDEC Solid State Technology Association et al. Failure mechanisms and models for semiconductor devices. *JEDEC Publication JEP122-C*, March 2006.

[56] P. N. Ji and Y. Aono. Colorless and directionless multi-degree reconfigurable optical add/drop multiplexers. In *Wireless and Optical Communications Conference*, pages 1–5, 2010.

[57] R.E Tarjan J.W. Suurballe. A quick method for finding shortest pairs of disjoint paths. *Networks*, 14(2):325–336, 1984.

[58] Ken-Ichi Kitayama, M. Sasaki, S. Araki, M. Tsubokawa, A. Tomita, K. Inoue, K. Harasawa, Y. Nagasako, and A. Takada. Security in photonic networks: threats and security enhancement. *IEEE/OSA Journal of Lightwave Technology*, 29(21):3210–3222, 2011.

[59] Arie M. C. A. Koster, Sebastian Orlowski, Christian Raack, Georg Baier, and Thomas Engel. Single-layer Cuts for Multi-Layer Network Design Problems. In Bruce Golden, Subramanian Raghavan, and Edward Wasil, editors, *Telecommunications Modeling Policy and Technology, also ZIB Report ZR-07-21*, chapter 1, pages 1–23. Springer, College Park, MD, U.S.A., 2008. Selected proceedings of the 9th INFORMS Telecommunications Conference.

[60] Arie M. C. A. Koster, Sebastian Orlowski, Christian Raack, Georg Baier, Thomas Engel, and Pietro Belotti. Branch-and-cut techniques for solving realistic two-layer network design problems. In Arie M. C. A. Koster and Xavier Muñoz, editors, *Graphs and Algorithms in Communication Networks*, chapter 3, pages 95–118. Springer, December 2009.

[61] G. Landi, N. Ciulli, J. Buysse, K. Georgakilas, M. Anastasopoulos, A. Tzanakaki, C. Develder, E. Escalona, D. Parniewicz, A. Binczewski, and B. Belter. A network control plane architecture for on-demand co-provisioning of optical network and it services. In *Future Network Mobile Summit (FutureNetw)*, July 2012.

[62] U. Mandal, M. Habib, Shuqiang Zhang, B. Mukherjee, and M. Tornatore. Greening the cloud using renewable-energy-aware service migration. *Network, IEEE*, 27(6):36–43, November 2013.

[63] S. S. Manson. Behavior of materials under conditions of thermal stress. *NACA Report 1170*, 1954.

[64] Sara Mattia. Solving survivable two-layer network design problems by metric inequalities. *Computational Optimization and Applications*, pages 1–26, 2010.

[65] Sara Mattia. A Polyhedral Study of the Capacity Formulation of the Multilayer Network Design Problem. *Networks*, 62(1):17–26, 2013.

[66] Ullrich Menne, Christian Raack, and Roland Wessäly. Impact of technology, architecture, and price trends on optimal multi-layer network designs. In *Proceedings of 16th International Conference on Optical Networking Design and Modeling (ONDM 2012)*, ONDM 2012, 2012, submitted. submitted.

[67] A. Muhammad, G. S. Zervas, N. Amaya, D. E. Simeonidou, and R. Forchheimer. Cost-efficient design of flexible optical networks implemented by architecture on demand. In *Proc. of IEEE/OSA Optical Fiber Communication Conference and Exposition*, OFC, 2014.

[68] Ajmal Muhammad, Georgios Zervas, Norberto Amaya, Dimitra Simeonidou, and Robert Forchheimer. Introducing Flexible and Synthetic Optical Networking: Planning and Operation Based on Network Function Programmable ROADMs. *IEEE/OSA Journal of Optical Communications and Networking*, 6(7):635–648, 2014.

[69] Avishek Nag, David B. Payne, and Marco Ruffini. N:1 protection design for minimising olts in resilient dual-homed long-reach passive optical network. In *Optical Fiber Communication Conference*, page Tu2F.7. Optical Society of America, 2014.

[70] K.C. Norris and A.H. Lanzberg. Reliability of controlled collapse interconnections. *IBM Journal of Research and Development*, 13(3):266–271, May 1969.

[71] E. Oki. *Linear Programming and Algorithms for Communication Networks A Practical Guide to Network Design, Control, and Management*. CRC Press, 2013.

[72] Sebastian Orlowski. *Optimal Design of Survivable Multi-layer Telecommunication Networks*. PhD thesis, Technische Universität Berlin, May 2009.

[73] Sebastian Orlowski, Arie M. C. A. Koster, Christian Raack, and Roland Wessäly. Two-layer Network Design by Branch-and-Cut featuring MIP-based Heuristics. In *Proceedings of the 3rd International Network Optimization Conference (INOC 2007)*, Spa, Belgium, 2007.

[74] Pablo Pavon-Marino and Jose-Luis Izquierdo-Zaragoza. On the role of open-source optical network planning. In *Optical Fiber Communication Conference*, page Th1E.1. Optical Society of America, 2014.

[75] Y. Peng, Z. Sun, S. Du, and K. Long. Propagation of all-optical crosstalk attack in transparent optical networks. *Optical Engineering*, 50(8):085002.1–3, 2011.

[76] Andrew Eugene Perkins. *Investigation and prediction of solder joint reliability for ceramic area array packages under thermal cycling, power cycling, and vibration environments*. PhD thesis, Georgia Institute of Technology, May 2007.

[77] Michal Pióro and Deepankar Medhi. *Routing, Flow, and Capacity Design in Communication and Computer Networks*. Morgan Kaufmann Publishers, 2004.

[78] Diane E. Hodges Popp, Andrew Mawer, and Gabriel Presas. Flip chip PBGA solder joint reliability: power cycling versus thermal cycling. *Motorola Semiconductor Products Sector, Austin, TX*, 2005.

[79] J. S. Arora R. T. Marler. The weighted sum method for multi-objective optimization: new insights. *Structural and Multidisciplinary Optimization*, 41(6):853–862, 2010.

[80] S. Ramamurthy and B. Mukherjee. Survivable wdm mesh networks. part i-protection. In *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 744–751 vol.2, Mar 1999.

[81] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee. Survivable wdm mesh networks. *J. Lightwave Technol.*, 21(4):870, Apr 2003.

[82] Franz Rambach, Beate Konrad, Lars Dembeck, Ulrich Gebhard, Matthias Gunkel, Marco Quagliotti, Laura Serra, and Víctor López. A Multilayer Cost Model for Metro/-Core Networks. *J. OPT. COMMUN. NETW.*, 5(3):210–225, 2013.

[83] Ridha Rejeb, Mark S. Leeson, and Roger Green. Fault and attack management in all-optical networks. *IEEE Communications Magazine*, 44(11):79–86, 2006.

[84] Ridha Rejeb, Mark S. Leeson, Carmen Mas Machuca, and Ioannis Tomkos. Control and management issues in all-optical networks. *Journal of Networks*, 5(2):132–139, 2010.

[85] D.A. Schupke and Franz Rambach. A link-flow model for dedicated path protection with approximative availability constraints. *Communications Letters, IEEE*, 10(9):679–681, Sept 2006.

[86] Qingya She, Xiaodong Huang, Qiong Zhang, Yi Zhu, and J.P. Jue. Survivable traffic grooming for anycasting in wdm mesh networks. In *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, pages 2253–2257, Nov 2007.

[87] Deepinder Sidhu, Raj Nair, and Shukri Abdallah. Finding disjoint paths in networks. *SIGCOMM Comput. Commun. Rev.*, 21(4):43–51, August 1991. ISSN 0146-4833.

[88] Jane M. Simmons. *Optical Network Design and Planning*. Springer Publishing Company, Incorporated, 1 edition, 2008. ISBN 0387764755, 9780387764757.

[89] L. Song, J. Zhang, and B. Mukherjee. A comprehensive study on backup-bandwidth reprovisioning after network-state updates in survivable telecom mesh networks. *Networking, IEEE/ACM Transactions on*, 16(6):1366–1377, Dec 2008.

[90] J.W. Suurballe. Disjoint paths in a network. *NETWORKS*, 4(2):125–145, 1974.

[91] Ward Van Heddeghem, Filip Idzikowski, Willem Vereecken, Didier Colle, Mario Pickavet, and Piet Demeester. Power consumption modeling in optical multilayer networks. *Photonic Network Communications*, 24(2):86–102, October 2012.

[92] Jean-Philippe Vasseur, Mario Pickavet, and Piet Demeester. *Network Recovery: Protection and Restoration of Optical, SONET-SDH, IP, and MPLS*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004. ISBN 012715051X.

[93] P. Wiatr, P. Monti, and L. Wosinska. Power savings versus network performance in dynamically provisioned WDM networks. *IEEE Communications Magazine*, 50(5): 48–55, May 2012. ISSN 0163-6804. doi: 10.1109/MCOM.2012.6194382.

[94] Pawel Wiatr, Jiajia Chen, Paolo Monti, and Lena Wosinska. Energy efficiency and reliability tradeoff in optical core networks. In *Proc. of the OFC, San Francisco, USA*, March 2014.

[95] Jing Wu and Michel Savoie. Recovery from control plane failures in the rsvp-te signaling protocol. *Computer Communications*, 34(16):1956–1967, October 2011.

[96] Yoshiyuki Yamada, Hiroshi Hasegawa, and Ken-Ichi Sato. Survivable Hierarchical Optical Path Network Design With Dedicated Wavelength Path Protection. *IEEE/OSA Journal of Lightwave Technology*, 29(21):3196–3209, 2011.

[97] Shengli Yuan and J.P. Jue. Dynamic lightpath protection in wdm mesh networks under wavelength continuity constraint. In *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, volume 3, pages 2019–2023 Vol.3, Nov 2004.

[98] J. Zhang, K. Zhu, and B. Mukherjee. Backup reprovisioning to remedy the effect of multiple link failures in wdm mesh networks. *Selected Areas in Communications, IEEE Journal on*, 24(8):57–67, Aug 2006.

[99] Mark Ziegelmann. *Constrained Shortest Paths and Related Problems - Constrained Network Optimization*. VDM Verlag, Saarbrücken, Germany, Germany, 2007. ISBN 3836446332, 9783836446334.