

OPTIMIZATION METHODS FOR OPTICAL LONG-HAUL  
AND ACCESS NETWORKS

SEYED MOHAMMAD KIAEI

A THESIS  
IN  
THE DEPARTMENT  
OF  
ELECTRICAL AND COMPUTER ENGINEERING

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY  
CONCORDIA UNIVERSITY  
MONTRÉAL, QUÉBEC, CANADA

SEPTEMBER 2011

© SEYED MOHAMMAD KIAEI, 2011

# CONCORDIA UNIVERSITY

## Engineering and Computer Science

This is to certify that the thesis prepared

By: **Mr. Seyed Mohammad Kiaei**

Entitled: **Optimization Methods for Optical Long-Haul and Access Networks**

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

**Doctor of Philosophy (Electrical and Computer Engineering)**

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

Signed by the final examining committee:

_____	Chair
_____	External Examiner
Dr. Pin-Han Ho	
_____	Examiner
Dr. Lata Narayanan	
_____	Examiner
Dr. Dongyu Qiu	
_____	Examiner
Dr. Yousef Shayan	
_____	Supervisor
Dr. Chadi Assi	
_____	Co-supervisor
Dr. Martin Maier	

Approved by \_\_\_\_\_

Dr. W. E. Lynch, Chair  
Department of Electrical and Computer Engineering

# Abstract

## Optimization Methods for Optical Long-Haul and Access Networks

Seyed Mohammad Kiaei, Ph.D.

Concordia University, 2011

Optical communications based on fiber optics and the associated technologies have seen remarkable progress over the past two decades. Widespread deployment of optical fiber has been witnessed in backbone and metro networks as well as access segments connecting to customer premises and homes. Designing and developing a reliable, robust and efficient end-to-end optical communication system have thus emerged as topics of utmost importance both to researchers and network operators. To fulfill these requirements, various problems have surfaced and received attention, such as network planning, capacity placement, traffic grooming, traffic scheduling, and bandwidth allocation. The optimal network design aims at addressing (one or more of) these problems based on some optimization objectives. In this thesis, we consider two of the most important problems in optical networks; namely the survivability in optical long-haul networks and the problem of bandwidth allocation and scheduling in optical access networks. For the former, we present efficient and accurate models for availability-aware design and service provisioning in  $p$ -cycle based survivable networks. We also derive optimization models for survivable network design based on  $p$ -trail, a more general protection structure, and compare its performance with  $p$ -cycles. Indeed, major cost savings can be obtained when the optical access and long-haul subnetworks become closer to each other by means of consolidation of access and metro networks. As this distance between long-haul and access networks reduces, and the need and expectations from passive optical access networks (PONs) soar, it becomes crucial to efficiently manage bandwidth in the access while providing

the desired level of service availability in the long-haul backbone. We therefore address in this thesis the problem of bandwidth management and scheduling in passive optical networks; we design efficient joint and non-joint scheduling and bandwidth allocation methods for multichannel PON as well as next generation 10Gbps Ethernet PON (10G-EPON) while addressing the problem of coexistence between 10G-EPONs and multichannel PONs.

*To my wonderful parents.*

# Acknowledgments

I would like to express my sincere gratitude to all people who supported me throughout the years of my PhD studies and beyond. I am truly indebted to my PhD supervisor, Dr. Chadi Assi, who not only was a dedicated and extremely helpful professor, but also supported me as an older brother during difficult situations. He initiated me to the beautiful world of optical networks and encouraged me to keep my integrity and strength during hard times. I would like to extend my honest appreciation to my co-supervisor, Dr. Martin Maier, who enlightened me through his vast knowledge of telecommunication networks and widely supported me during the preparation of this thesis.

During this thesis, I collaborated with many scholars and researchers in the field of optimization and communication networks. These collaborations yielded to authoring high-quality and well received journal and conference papers. In this matter, I am grateful to my co-authors, Mr. Amin Ranjbar, Dr. Samir Sebbah, Dr. Brigitte Jaumard, Dr. Antone Cerny, and Mr. Lehan Meng. Furthermore, I was granted a warm and friendly atmosphere in our research lab at Concordia University throughout my PhD. I would like to express my warm thanks to all my office mates, especially Mr. Mohammad Faisal Uddin. I am also grateful to all my friends in Montreal, and other places who supported me during my education and my personal life.

Last but not least, my deepest appreciation goes to my parents who raised me with love and highest possible devotion. I could never achieve my ambitions without their encouragement, understanding, support, and true love. I would like to dedicate

this thesis to my wonderful mother who thought me that great expectations can come true by means of hard effort and perseverance. This thesis is also dedicated to my kindhearted father who has always brought confidence to my life through his utmost support and sympathy.

# Contents

<b>List of Figures</b>	<b>xii</b>
<b>List of Tables</b>	<b>xv</b>
<b>List of Publications</b>	<b>xvii</b>
<b>List of Acronyms</b>	<b>xviii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	1
1.2 Problem Statements and Motivations . . . . .	4
1.2.1 Mesh-Based Survivable Network Design . . . . .	4
1.2.2 Resource Management in Passive Optical Networks . . . . .	6
1.3 Thesis Objectives . . . . .	8
1.4 Thesis Contributions . . . . .	9
1.5 Thesis Outline . . . . .	11
<b>2 Background and Related Work</b>	<b>12</b>
2.1 Survivability in Optical Networks . . . . .	12
2.1.1 Basic Classification of Survivability Schemes . . . . .	13
2.1.2 Ring-based vs. Mesh-based Survivability . . . . .	15
2.2 $p$ -Cycle Protection Method . . . . .	17
2.2.1 The Concept of $p$ -Cycle . . . . .	18

2.2.2	Path-Protecting $p$ -Cycles . . . . .	19
2.2.3	Optimal Spare Capacity Design with $p$ -Cycles . . . . .	23
2.2.4	Multiple Failure Survivability with $p$ -Cycles . . . . .	27
2.2.5	Availability Analysis of $p$ -Cycle-Based Networks . . . . .	30
2.3	Survivable Network Design based on $p$ -Trail . . . . .	34
2.4	Optical Access Networks . . . . .	35
2.4.1	Various PON Architectures and Standards . . . . .	37
2.4.2	Scheduling and Bandwidth Allocation in PONs . . . . .	42
<b>3</b>	<b>Availability-Aware Service Provisioning in <math>p</math>-Cycle-Based Networks</b>	<b>47</b>
3.1	Availability Analysis of $p$ -Cycle Based Networks . . . . .	48
3.2	Improved Availability Modeling . . . . .	50
3.2.1	Overcounting of Spans in the Protection Domain . . . . .	51
3.2.2	Exhaustive Enumeration of the Dual Failure Scenarios . . . . .	53
3.3	The ApC Model . . . . .	55
3.3.1	Minimum Spare Capacity Optimization Model . . . . .	56
3.3.2	Linearizing the Quadratic terms . . . . .	59
3.4	ILP Solution and Scalability Issues . . . . .	61
3.4.1	Selection of the $p$ -cycles . . . . .	61
3.4.2	ILP Heuristic Techniques . . . . .	61
3.4.3	Incremental Optimality Gap . . . . .	62
3.4.4	Unavailability Overestimation . . . . .	62
3.5	Numerical Results . . . . .	63
3.5.1	Capacity Efficiency . . . . .	64
3.5.2	Design Complexity . . . . .	66
3.5.3	Accuracy . . . . .	68
3.5.4	Tradeoff between MTTR and Redundancy . . . . .	69
3.6	Conclusion . . . . .	71

<b>4</b>	<b>Network Protection Design Models Using Pre-Cross-Connected Trails</b>	<b>73</b>
4.1	Motivation and Problem Statement . . . . .	74
4.2	ILP Formulation . . . . .	77
4.3	A Column Generation Design Model . . . . .	80
4.3.1	Master Problem . . . . .	81
4.3.2	Pricing Problem . . . . .	82
4.3.3	Discussion . . . . .	84
4.4	Numerical Results . . . . .	88
4.5	Conclusion . . . . .	96
<b>5</b>	<b>Scheduling and Grant Sizing Methods for WDM PONs</b>	<b>97</b>
5.1	Network Architecture . . . . .	98
5.2	Motivation and Problem Statement . . . . .	99
5.3	Mathematical Formulation . . . . .	102
5.3.1	Non-Joint Grant Scheduling . . . . .	102
5.3.2	Joint Scheduling and Grant Sizing . . . . .	104
5.4	Solving The Joint Problem Using Tabu Search . . . . .	109
5.5	Numerical Results . . . . .	111
5.5.1	Different Network Instances . . . . .	112
5.5.2	Different Traffic Loads . . . . .	114
5.5.3	Packet Level Simulation . . . . .	116
5.6	Conclusion . . . . .	117
<b>6</b>	<b>Scheduling and Bandwidth Allocation of 10G-EPON co-Existing with WDM-PON</b>	<b>120</b>
6.1	Network Architecture . . . . .	121
6.2	Motivation and Problem Statement . . . . .	123

6.3	Scheduling and Bandwidth Allocation for 10G-TDM and 1G-WDM	
	ONUs . . . . .	125
6.3.1	Bandwidth Allocation . . . . .	125
6.3.2	Delay Analysis . . . . .	127
6.3.3	ILP Model . . . . .	128
6.4	A Tabu Search Heuristic for Solving the Scheduling Problem . . . . .	132
6.5	Numerical Results . . . . .	134
6.5.1	Evaluation of Different Methods . . . . .	135
6.5.2	Upgrading ONUs from 1Gbps to 10Gbps . . . . .	135
6.5.3	Packet Level Simulation . . . . .	138
6.6	Conclusion . . . . .	140
<b>7</b>	<b>Conclusion and Future Directions</b>	<b>144</b>
7.1	Conclusions . . . . .	144
7.2	Future Work . . . . .	146

# List of Figures

1.1	Internet infrastructure hierarchy consisting of access, metro, and long-haul networks. . . . .	3
1.2	Tree topology of passive optical networks . . . . .	7
2.1	Basic operation of $p$ -cycles for protection of on-cycle and straddling link failures . . . . .	18
2.2	A set of three mutually disjoint working routes and their corresponding shared backup paths. . . . .	20
2.3	A FIPP $p$ -cycle is protecting a set of five mutually disjoint working paths. . . . .	23
2.4	Three categories of dual failure scenarios which result in service outage	31
2.5	Protection schemes with and without branch points . . . . .	35
2.6	Waveband allocation in 1G- and 10G-EPON . . . . .	40
2.7	A typical network architecture for the co-existence of 1G- and 10G-EPONs . . . . .	41
2.8	Timing diagram of IPACT . . . . .	43
3.1	Spans traverse two protection domains . . . . .	51
3.2	New category of dual failure scenarios which result in service outage .	54
3.3	Dual failure in two adjacent on-cycle spans . . . . .	54
3.4	Average cycle length vs. minimum required availability for the two network instances with 3 demands on each node pair (ApC-over). . .	67

3.5	Precision gap of three models for different values of required service availability in 9n17s network with 2 demands on each node pair. . . .	69
3.6	Resource redundancy for varying values of physical unavailability of each span; the minimum service availability 99.99% . . . . .	71
3.7	Resource redundancy for varying values of physical unavailability of each span; the minimum service availability is set to 99.999% . . . .	72
4.1	An example of a network protected by a set of $p$ -trails . . . . .	75
4.2	A simple $p$ -cycle constructed by merging the underlying elementary simple trails . . . . .	75
4.3	Survivable network design based on $p$ -cycles and $p$ -trails . . . . .	76
4.4	A non-simple cycle whose protection capacity cannot be explored by merging the underlying simple trails . . . . .	85
4.5	Network instances considered in [1] . . . . .	90
4.6	Additional selected networks for numerical results . . . . .	91
4.7	Performance evaluation by varying the distribution of cost per link . .	92
4.8	Performance evaluation by varying the distribution of the number of demands per link . . . . .	95
5.1	Evolutionary upgrade from TDM PON to WDM PON . . . . .	99
5.2	An illustrative example for comparing different scheduling methods .	101
5.3	Makespan for different experiment groups . . . . .	115
5.4	Average channel utilization for different experiment groups . . . . .	116
6.1	Coexistence of 10G-TDM and 1G-WDM PONs . . . . .	122
6.2	An illustrative example for comparing different scheduling frameworks	141
6.3	Dynamics of bandwidth allocation . . . . .	142
6.4	Results of upgrading scenarios. (a) network traffic load (Gb/s), (b) makespan ( $\mu$ sec), (c) Expected maximum delay (msec), (d) average channel utilization . . . . .	142

6.5	Comparison of our Tabu method with online NASC and NJS-TCT . . .	143
-----	--	-----

# List of Tables

3.1	Comparing the Spans Belonging to Different Availability Sets in Our Model with the Model of [2] . . . . .	52
3.2	Comparing the Redundancy in our models and prior work in [2] for different Availabilities on network instance 9n17s . . . . .	65
3.3	Comparing the Redundancy on COST239 network . . . . .	66
3.4	Comparison of the Complexity of Different Models for One Network Instance . . . . .	68
3.5	The amount of overestimation in ApC-over for 9n17s network with 3 demands on each node pair and minimum availability of 99.9988% . . . . .	70
4.1	Evaluating the ILP model on different network instances . . . . .	80
4.2	Characteristics of Sample Networks . . . . .	89
4.3	Comparison of different protection schemes . . . . .	91
4.4	Size of ILP Models . . . . .	93
5.1	Supported channels of ONUs and their corresponding scheduling order	100
5.2	Network Parameters . . . . .	111
5.3	Makespan ( <i>msec</i> ) when each ONU has a traffic load in the interval of $[0.25B_{min}, 2.5B_{min}]$ . . . . .	112
5.4	Average channel utilization (%) when each ONU has a traffic load in the interval of $[0.25B_{min}, 2.5B_{min}]$ . . . . .	113
5.5	CPU time (in sec) for different scheduling algorithms . . . . .	114
5.6	Average queuing delay ( <i>msec</i> ) . . . . .	118

5.7	Average packet loss rate (%) . . . . .	119
6.1	Network Parameters . . . . .	134
6.2	Load distribution for 1G- and 10G-ONUs in different experiments . .	134
6.3	Makespan ( $\mu\text{sec}$ ) . . . . .	136
6.4	Average Channel Utilization (percentage). . . . .	136
6.5	Results of the joint Tabu method for a network with 6 1G WDM channels, and a total of 40 ONUs, when the 10G channel is shared with 1G-ONUs which are gradually upgraded to 10G. . . . .	137
6.6	Average Queuing Delay ( $\mu\text{sec}$ ) . . . . .	139
6.7	Average Packet Loss Rate (percentage) . . . . .	139

# List of Publications

## Journals

1. Mohammad S. Kiaei, Chadi Assi, and Brigitte Jaumard, “A survey on the  $p$ -cycle protection method,” *IEEE Communications Surveys and Tutorials*, vol. 11, no. 3, pp. 53–70, Third Quarter 2009.
2. Mohammad S. Kiaei, Amin Ranjbar, Brigitte Jaumard, and Chadi Assi, “An improved analysis for availability-aware service provisioning in  $p$ -cycle-based mesh networks,” *IEEE/OSA Journal of Lightwave Technology*, vol. 27, no. 20, pp. 4424–4434, Oct. 2009.
3. Mohammad S. Kiaei, Lehan Meng, Chadi Assi, and Martin Maier, “Efficient scheduling and grant sizing methods for WDM PONs,” *IEEE/OSA Journal of Lightwave Technology*, vol. 28, no. 13, pp. 1922–1931, July 2010.
4. Mohammad S. Kiaei, Lehan Meng, Chadi Assi, and Martin Maier, “On the co-existence of 10G-EPONs and WDM-PONs: A scheduling and bandwidth allocation approach,” *IEEE/OSA Journal of Lightwave Technology*, to appear.
5. Mohammad S. Kiaei, Samir Sebbah, Antone Cerny, Hamed Alazemi, and Chadi Assi, “Efficient network protection design models using pre-cross-connected trails,” *IEEE Transactions on Communications*, to appear.

## Conference Proceedings

1. Mohammad S. Kiaei, Amin Ranjbar, Caroline Rocha, Brigitte Jaumard, and Chadi Assi, “Improved availability models for  $p$ -cycle-based network design,” in *Proc., 7th International Workshop on the Design of Reliable Communication Networks (DRCN 2009)*, Washington, DC, USA, Oct. 2009.
2. Mohammad S. Kiaei, Lehan Meng, Chadi Assi, and Martin Maier, “Efficient joint scheduling and grant sizing in multichannel WDM PONs,” in *Proc., IEEE International Conference on Communications (ICC)*, Cape Town, South Africa, May 2010.
3. Mohammad S. Kiaei, Lehan Meng, Chadi Assi, and Martin Maier, “Joint scheduling and bandwidth allocation methods for 10G-EPON and beyond,” *IEEE International Conference on Communications (ICC)*, Kyoto, Japan, June 2011.
4. Mohammad S. Kiaei, Samir Sebbah, Antone Cerny, Hamed Alazemi, and Chadi Assi, “Survivable Network Design Models based on Pre-Cross-Connected Trails,” *IEEE International Conference on Computer Communication Networks*, Maui, Hawaii, USA, Aug. 2011.

# List of Acronyms

<b>AE</b>	– A Priori Efficiency
<b>APON</b>	– ATM Passive Optical Network
<b>APS</b>	– Automatic Protection Switching
<b>ATM</b>	– Asynchronous Transfer Mode
<b>BLSR</b>	– Bi-directional Line Switched Ring
<b>BPON</b>	– Broadband Passive Optical Network
<b>CBR</b>	– Constant Bit-Rate
<b>CG</b>	– Column Generation
<b>CM</b>	– Cable Modem
<b>CW</b>	– Continuous Wave
<b>DBA</b>	– Dynamic Bandwidth Allocation
<b>DCC</b>	– Digital Cross Connect
<b>DSL</b>	– Digital Subscriber Line
<b>EPON</b>	– Ethernet Passive Optical Network
<b>FEC</b>	– Forward Error Correction
<b>FIPP</b>	– Failure Independent Path Protecting $p$ -cycle
<b>FTTB</b>	– Fiber to the Building
<b>FTTC</b>	– Fiber to the Curb
<b>FTTH</b>	– Fiber to the Home
<b>FTTP</b>	– Fiber to the Premises

<b>GPON</b>	– Gigabit-capable Passive Optical Network
<b>HDTV</b>	– High-Definition Television
<b>ILP</b>	– Integer Linear Programming
<b>IPACT</b>	– Interleaved Polling With Adaptive Cycle Time
<b>IPTV</b>	– Internet Protocol television
<b>ISCG</b>	– Inter-Scheduling Cycle Gap
<b>ISP</b>	– Internet Service Provider
<b>JIT</b>	– Just in Time
<b>JSBA</b>	– Joint Scheduling and Bandwidth Allocation
<b>LAN</b>	– Local Area Networks
<b>LARNET</b>	– Local Access Router Network
<b>LFJ</b>	– Least Flexible Job
<b>LP</b>	– Linear Programming
<b>LPT</b>	– Largest Processing Time
<b>LR-PON</b>	– Long-Reach PON
<b>LSP</b>	– Label Switched Path
<b>MAN</b>	– Metropolitan Area Network
<b>MFS</b>	– Multi Failure Survivability
<b>MIP</b>	– Mixed Integer Programming
<b>MJSBA</b>	– Modified Joint Scheduling and Bandwidth Allocation
<b>MPCP</b>	– Multi-Point Control Protocol
<b>MRCP</b>	– Multi Restorability Capacity Placement
<b>MTTF</b>	– Mean Time To Failure
<b>MTTR</b>	– Mean Time To Repair
<b>NASC</b>	– Next Available Supported Channel
<b>NJS</b>	– Non-Joint Scheduling
<b>OADM</b>	– Optical Add Drop Multiplexer

<b>OLT</b>	– Optical Line Terminal
<b>ONU</b>	– Optical Network Unit
<b>OXC</b>	– Optical Cross-Connect
<b>P2P</b>	– Peer-to-Peer
<b>PM</b>	– Parallel Machine
<b>PON</b>	– Passive Optical Network
<b>PR</b>	– Path Restoration
<b>PSTN</b>	– Public Switched Telephone Networks
<b>QoS</b>	– Quality of Service
<b>RFS</b>	– Routing of Flows and Slacks
<b>RITENET</b>	– Remote Interrogation of Terminal Network
<b>RSOA</b>	– Reflective Semiconductor Optical Amplifier
<b>RTT</b>	– Round-Trip Time
<b>SBPP</b>	– Shared Backup Path Protection
<b>SCA</b>	– Spare Capacity Assignment
<b>SDH</b>	– Synchronous Digital Hierarchy
<b>SEACP</b>	– Selectively Enhanced Availability Capacity Placement
<b>SG-EPON</b>	– STARGATE EPON
<b>SHR</b>	– Self-Healing Ring
<b>SLA</b>	– Service Level Agreement
<b>SNR</b>	– Signal to Noise Ratio
<b>SONET</b>	– Synchronous Optical Network
<b>SPT</b>	– Shortest Processing Time
<b>SUCCESS</b>	– Stanford University Access Network
<b>TDM</b>	– Time Division Multiplexing
<b>TDMA</b>	– Time Division Multiple Access
<b>UPSR</b>	– Unidirectional Path-Switched Ring

- WAN** – Wide Area Network
- WDM** – Wavelength Division Multiplexing
- WDMA** – Wavelength Division Multiple Access

# Chapter 1

## Introduction

### 1.1 Overview

Optical fiber is a key enabling technology in modern telecommunication networks, extending from backbone networks to access segments and all the way to the customer premises. This technology provides network operators with the critical capacity needed for supporting emerging services with intensive bandwidth requirements such as peer to peer multimedia services, telemedicine, video conferencing, storage, and voice/video over IP.

Transport networks based on fiber optic can bring fast and high-quality services to end users. As a transmission medium, optical fiber brings many advantages such as low loss, light weight, electromagnetic immunity, high bandwidth and low cost. The most attractive feature of optical fiber is its extremely large capacity, in the order of a few Tera-bit per second, which is provided at low cost. In particular, wavelength division multiplexing (WDM) technology enables a single optical fiber to offer point-to-point capacities in the order of hundreds of Tera-bit per second in total. Optical fiber is a reliable medium, because it is immune to electromagnetic disturbances and it can cover long distances without a need for electrical regeneration. Today, the network topology is becoming more of an all-optical network, incorporating network

elements such as optical add/drop multiplexer (OADM) and optical cross-connect (OXC).

In order to provide a reliable and continuous end-to-end service in the transport network, it is more convenient to view the whole network as consisting of subnetworks and determine and solve the major issues pertaining to each subnetwork. Generally, a public network can be partitioned into subnetworks according to geographic (or operational) boundaries in order to reflect the differences in demand distribution, cost structures and topological layout. One common way of partitioning is dividing a network into access, metropolitan (or metro), and core (or long-haul) subnetworks as depicted in Figure 1.1. In an access subnetwork, most demands originate at remote switching offices and customer premises and terminate back at a main switching office (or hub). An access network consists of various premises such as residential digital subscriber line (DSL) or cable modems, regional Internet service providers (ISP), corporate enterprise clients, and public switched telephone networks (PSTN). A metro subnetwork connects main switching offices within a metropolitan area and demands are typically more uniformly distributed. Because the distances in access and metro subnetworks are typically less than 25 to 50 km, nodal equipment costs usually dominate total network costs. Long-haul subnetworks, on the other hand, usually connect metropolitan areas to each other or interconnect with other long-haul networks, enabling seamless and efficient inter-city and international connectivity. Long-haul networks carry a lot more data than any other type of network, and cover much greater distances, which can reach hundreds or thousands of kilometers. Therefore, distance-related costs for cable installation, amplifiers, and regenerators can dominate the total cost [3]. SONET (synchronous optical network) or SDH (synchronous digital hierarchy) are the most frequent transport technologies used in long-haul networks. The combination of long-haul and metropolitan sub-networks is usually called backbone network. The backbone of the telecommunication networks, usually referred

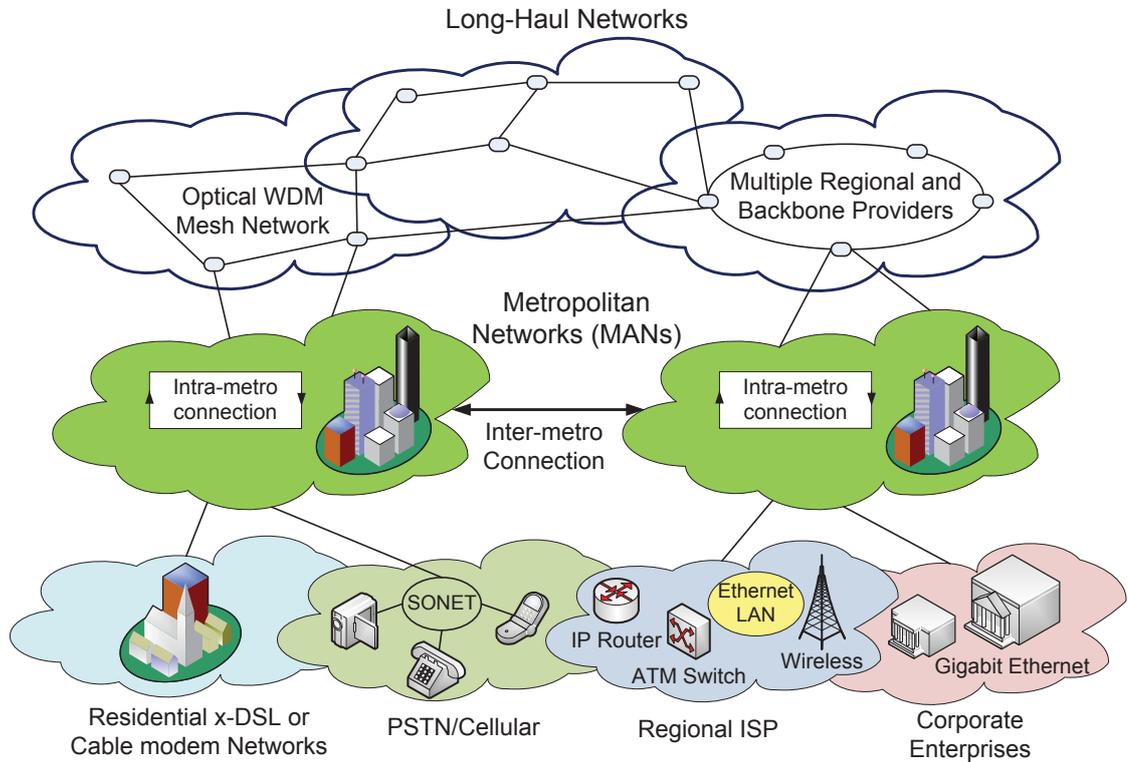


Figure 1.1: Internet infrastructure hierarchy consisting of access, metro, and long-haul networks.

to as the core or the transport network, is the heart of all large network provider operations. This fiber “highway” is constantly evolving and becoming bigger, faster, and more complex.

In recent years, WDM systems have seen wide deployment in long-haul service provider networks, and are increasingly being deployed in metro networks and for enterprise data center connectivity applications [4]. Several key issues have to be addressed in the design and development of different optical subnetworks such as reduction of the network downtime, improvement of network reliability, reduction of fiber optical maintenance cost, and improvement of service level agreement (SLA) management. To fulfill these requirements, various problem statements have emerged such as network planning, capacity placement, provisioning, survivability, routing,

wavelength assignment, traffic grooming, traffic scheduling, and bandwidth allocation. An optimal network design aims at addressing (one or more of) these problems based on some optimization objectives. In this thesis, we consider two of the most important problems in optical networks; namely the survivability of optical long-haul networks and the problem of bandwidth allocation and scheduling in optical access networks.

## **1.2 Problem Statements and Motivations**

### **1.2.1 Mesh-Based Survivable Network Design**

In recent years, the number of critical business users, which rely on transport networks has been growing very rapidly. Thus, any interruption of service for even short periods of time may lead to disastrous consequences. Companies cannot afford the business risk and reputation loss of having their networks down even for very short periods. Therefore, preventing service interruption and minimizing the loss of service, have become problems of utmost importance and must be addressed carefully when designing reliable telecommunication networks, particularly for optical transport networks with very large data transfer rates.

Optical communication is a cable-based technology which is either buried underground or on poles or lies at the bottom of the ocean. In all of these cases, the optical connection is dramatically vulnerable to cable cuts, especially in metro and long-haul networks, where hundreds of kilometers of fiber are being utilized. These facts gave researchers motivation to work on improving the “survivability” of these networks. In general, survivability refers to the ability of a network to provide continuous services in presence of failures. Given that survivability in the optical layer is more reliable and cost-effective than in other layers [5], substantial studies have been devoted to the survivability of optical networks for the past two decades [6–8]. Using a reliable

and survivable design concept, network operators can drastically reduce the network downtime and improve network reliability and SLA management.

It should be noted that networks that are fully restorable to single fiber cuts might be unable to fully recover against higher-order failure combinations. In other words, making a network fully restorable for single failures is no guarantee that the “availability” of the service in the occurrence of higher-order failures will be 100%. The availability of a network is determined by the duration of service outage, which is caused by multiple concurrent failures. Former studies declare that dual link-failures are the main contributors to service unavailability in long-haul mesh networks [9]. In fact, considering only dual link-failures is sufficient to obtain a good estimate of the expected service availability. Methods such as post-failure reconfiguration and pre-failure provisioning of additional protection capacity have been considered to add another level of protection against dual-failures. Another alternative is the network design with “availability awareness.” In this approach, the amount of required service availability is taken into account while provisioning the spare capacity for single-failure restorability.

One of the most efficient methods for the design of mesh-based survivable networks is the pre-configured protection cycle, known as  $p$ -cycle [10]. The basic idea of  $p$ -cycle is to build the protection paths by utilizing the concept of fully pre-cross-connected linear segments [11].  $p$ -Cycles gather the desired characteristics of mesh-based and ring-based protection methods, i.e., achieving the speed of line-switched self-healing rings while having the capacity efficiency of a mesh-restorable network.  $p$ -Cycle was initially introduced as a “link-protecting” scheme whose objective is to guarantee the recovery of affected service in the event of any single link failure. Later,  $p$ -cycle was extended for path protection by introducing failure independent path protecting  $p$ -cycles (FIPP) [12]. The concept of  $p$ -cycle was later generalized in [13], by observing that the high speed protection capability of rings and  $p$ -cycles is not due to their

circular topology but rather because their protection routes are pre-cross-connectable. This generalization leads to the definition of pre-cross-connected trails or  $p$ -trails. Similar to  $p$ -cycles,  $p$ -trails achieve the speed of rings with the efficiency of mesh, but they are more flexible than  $p$ -cycles. Theoretically, a  $p$ -trail-based network design can yield a better capacity efficiency compared to the  $p$ -cycle solution, because  $p$ -cycles can be viewed as a special case of  $p$ -trails.

### 1.2.2 Resource Management in Passive Optical Networks

Consolidation of optical access and metro networks is a success story in next-generation passive optical networks (PONs). Long-haul network-based companies aim to get progressively closer to the end customers. Major cost savings can be obtained when the optical access and long-haul subnetworks become closer to each other by means of consolidation of access and metro networks. The access and metro networks can be combined into one through the use of an extended backhaul fiber, possibly 100 km in length to incorporate protection paths and mechanisms, used with a PON [14]. Significant cost reduction can be obtained as the legacy SONET/SDH rings are replaced with a single backhaul fiber. Terminating at a core node, the combined access and backhaul network can potentially remove the local exchange site [14].

As the distance between long-haul and access subnetworks decreases, it is becoming more crucial to tackle the optimal resource management in optical access networks while providing a desired level of availability in backbone long-haul optical networks. This intrigues us to address the problem of bandwidth management and scheduling in PONs.

PONs have become increasingly popular due to their capability of building efficient broadband access networks that enable the support of a wide range of new services and applications such as triple play, video on demand, video conferencing, peer-to-peer (P2P) audio/video file sharing, etc. A PON generally consists of one optical

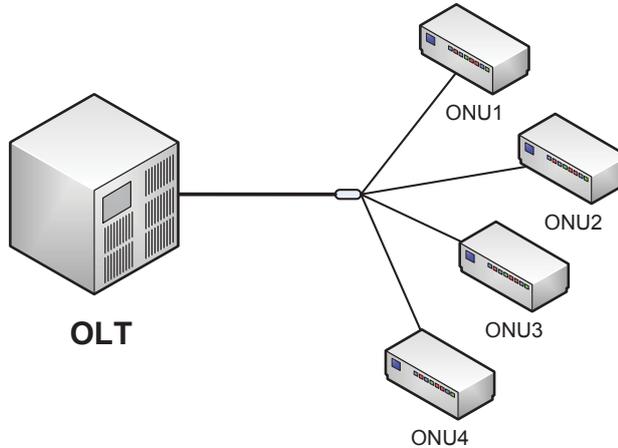


Figure 1.2: Tree topology of passive optical networks

line terminal (OLT) located at the central office of the service provider, which is connected to several optical network units (ONUs) located either at the end-user location (FTTH and FTTB), or at the curb, resulting in a fiber-to-the-curb (FTTC) architecture. The connection between OLT and ONUs can be realized in different topologies such as tree, ring, or bus, among which the tree topology is the most popular one as depicted in Figure 1.2 [15].

In the downstream direction from OLT to ONUs, the PON is a point-to-multipoint medium. Conversely, in the upstream direction it is a multipoint-to-point network, where the ONUs share the same fiber but the upstream optical signal is not received by the ONUs. Thus, time division multiplexing (TDM) or WDM should be used in order to avoid collision in the upstream direction [15]. A centralized medium access control (MAC) protocol is required at the OLT to arbitrate ONU's upstream transmissions. In addition, the OLT performs dynamic bandwidth allocation (DBA) for grant sizing and bandwidth allocation to each ONU.

Thanks to the WDM technology, significant progress has been made in terms of cost reduction in multichannel upgrades of PONs [16]. By employing WDM, a PON can support multiple wavelengths in either or both upstream and downstream directions. This way, existing Ethernet PON (EPON) can be upgraded to multi-channel

PON, which is referred to as WDM PON. EPON technology has, however, been recently extended to 10G-EPON in order to provide ten-fold data rates of 10 Gbps. 10G-EPON has emerged as a promising candidate for next-generation high data rate access systems [17]. The drivers behind 10G-EPON are mainly to serve business users and bandwidth-intensive residential customers that require high bandwidth services such as HDTV. Besides, in fiber to the building (FTTB) type of topologies, 10G-EPON with enhanced split ratio can lower the expenditures of carriers. 10G-EPON has the advantage through its coexistence attribute to allow smooth and gradual EPON upgrades to carriers. 10G-EPON can reduce expenditures for carriers, specifically in FTTB applications.

The bandwidth allocation and management problem is a key design issue for every PON system. This becomes more critical in WDM PON and 10G-EPON in order to exploit their full benefits of the multichannel and multi-rate upgrades. Compared to conventional PONs, WDM PON and 10G-EPON require more sophisticated grant scheduling and bandwidth allocation mechanisms for handling bandwidth demands of multiple ONUs such that efficient bandwidth utilization is achieved.

### 1.3 Thesis Objectives

In this thesis, we consider two of the most important problems in optical networks; namely 1) the design of reliable optical long-haul transport networks and 2) the problem of efficient bandwidth allocation and scheduling in optical access networks. For the former, we present accurate and effective design models for availability-aware service provisioning in  $p$ -cycle based survivable networks. We also construct optimization models for survivable network design based on  $p$ -trails and compare its performance with  $p$ -cycles. For the latter problem, we design efficient joint and non-joint scheduling and bandwidth allocation methods for evolutionary upgraded multichannel PON as well as for 10G-EPON, while addressing the problem of coexistence

between 10G-EPONs and multichannel PONs.

## 1.4 Thesis Contributions

- The shortcomings of the existing models for availability-aware  $p$ -cycle networks are first highlighted. Then, a more accurate model is derived after addressing those subtle issues. In the proposed model, all dual-failure scenarios, which may lead to service outage on the routed demand, are exhaustively enumerated. It is shown that a very meticulous analysis must be done on each protection domain traversed by the service path so that an overestimation of the unavailability of the service is avoided. The resulting model is hence a more accurate model, but with less scalability. Accordingly, several techniques are introduced to address the scalability issues of the proposed model. Results show that, in spite of not being able to solve optimally the proposed model, very good estimation of the network unavailability can be attained.
- An in-depth study is carried out to investigate the capability of  $p$ -trails in protecting traffic demands in a mesh-based survivable network. By taking the sharing capability of  $p$ -trails into account, optimization models are introduced to verify the remarkable efficiency of  $p$ -trails. Two ILP models are derived for survivable network design using  $p$ -trails. In the first model, the optimal solution is obtained from a candidate set constructed by exhaustive enumeration of all simple trails. It will be shown that the size of this ILP model, and therefore the computation time, becomes prohibitively large, making the model impractical for larger network instances. Therefore, to overcome this scalability issue, a better model for this complex optimization problem is developed using a primal-dual decomposition of the original problem based on the column generation (CG) optimization method. This design approach is shown to be very scalable,

as opposed to other prior  $p$ -trail design methods; further, results show that  $p$ -trails are more efficient than  $p$ -cycles in terms of resource redundancy in the network.

- In most of the previous studies on scheduling and bandwidth allocation in PONs, the grant sizing and grant scheduling subproblems have been considered separately, which may not achieve optimal network performance. The non-joint problem will be revisited and a more efficient ILP model will be derived when the bandwidth allocation is pre-determined. Then, the problem of joint grant sizing and scheduling for multichannel access networks will be investigated. The performance of the joint model will be compared to that (non-joint) of previous studies. Since the joint model is shown to be hard to solve, except for small network instances, a Tabu search heuristic will be introduced for achieving near optimal solutions in a reasonable amount of time.
- The problem of optimal scheduling and bandwidth allocation in next generation 10G-EPON coexisting with 1G WDM-PONs will be elaborated. First, a network architecture for supporting the coexistence will be introduced. Then, an ILP model will be derived for offline joint scheduling and bandwidth assignment for 10G-TDM and 1G-WDM ONUs. The aim is to develop efficient bandwidth allocation and scheduling algorithms for this system with multi-rate ONUs. Based on the choice of wavelength channels, the OLT may use separate or the same DBA modules for 1G- and 10G-PONs. To address this fact, two scheduling scenarios will be studied where the 10G TDM channel is either shared between 1G- and 10G-ONUs, or it is dedicated to 10G-ONUs. The tradeoff in terms of delay, scheduling length, and channel utilization will be explored, when separate or the same DBA modules are used for 1G- and 10G-ONUs. To address the scalability of the ILP model, a Tabu Search based heuristic will be introduced for obtaining near-optimal solutions in remarkably

shorter computation times.

Different tools and methods are employed to achieve the objectives and to evaluate the proposed design models. The optimization models proposed in this thesis are implemented in C++, using the “CPLEX Concert Technology” and their solutions are obtained by using the solver CPLEX 11.0.1 [18]. Heuristic methods such as Tabu search are also implemented in C++. To study the performance of the proposed scheduling methods in PONs, we carry out packet-level simulation using OMNet++, which is a discrete event simulator [19].

## 1.5 Thesis Outline

The rest of the thesis is organized as follows. Chapter 2 presents the background and reviews the related work in the fields investigated throughout this thesis. In Chapter 3, a more accurate model is presented for availability-aware service provisioning in  $p$ -cycle based networks. Survivable network design models based on  $p$ -trails are presented in Chapter 4. In Chapter 5, we investigate the problem of scheduling and bandwidth allocation for evolutionary upgraded WDM PONs. Chapter 6 presents our resource management methods for the coexistence of 10G-EPONs 1G-WDM PONs. Finally, Chapter 7 summarizes our conclusions and presents some future research directions.

# Chapter 2

## Background and Related Work

In this chapter, the background and the literature survey for topics investigated throughout this thesis are presented. This chapter is structured as follows. Section 2.1 presents the concept and basic classification of survivability schemes in optical networks. In Section 2.2, the  $p$ -cycle protection method is explained, and various network design methods based on  $p$ -cycles are surveyed. Survivable network design based on  $p$ -trails is introduced in Section 2.3. Finally, Section 2.4 presents an overview of optical access networks along with an explanation of various architectures for passive optical networks and their design schemes.

### 2.1 Survivability in Optical Networks

In studying network survivability, two basic types of network element failures are normally considered: link and node failure. Link failure is usually caused by cable cuts, while node failure is due to equipment failure at network nodes. Another less considered type of failure in WDM optical networks is channel failure, which is usually caused by the failure of transmitting and/or receiving equipment operating on that channel [3]. The performance of different survivability schemes is often evaluated using different metrics such as complexity, speed and capacity efficiency. The

overall complexity of a survivability method can be assessed in two ways. One is operational complexity which is measured by the required attempt for utilizing the backup resources. The other is design complexity which is determined by the computational cost in the mathematical model of the considered method. The speed of a survivability method is determined by the amount of time required for the activation of spare capacity upon occurrence of a failure. “Capacity efficiency” is defined as the reciprocal of the redundancy. Generally, in studying optical networks, the “geographical redundancy” is defined as the ratio of “protection cost” to “working cost”. The protection (working) cost is the sum of required spare (working) channels weighted by a coefficient representing either the distance of a link or the cost per channel on the link [3].

### **2.1.1 Basic Classification of Survivability Schemes**

Survivability schemes in optical networks can be classified under two general categories: protection and restoration [8, 20]. “Protection” is a pre-planned and pre-configured scheme, where some resources are reserved for recovery from failures at either connection setup or network design time, and kept idle when there is no failure. The advantage is that it provides fast and 100 percent failure recovery, but it is not efficient in terms of capacity. The other category of survivability schemes is “Restoration” where the spare capacity in the network is dynamically discovered to recover the affected services upon occurrence of a failure. In other words, unlike the protection schemes, there are no reserved resources for recovery at the time of connection establishment, and recovery is achieved by using the available resources such as fibers, wavelengths, and switches when the failure occurs. Therefore, the restoration time is usually longer, and 100 percent service recovery cannot be guaranteed because sufficient spare capacity may not be available at the time of failure.

But, it would be typically more efficient than protection schemes in terms of capacity. Most studies in the field of survivability in WDM optical networks are focused on protection rather than restoration schemes. Protection against single-link failures in WDM networks can be divided into two main groups; link-based protection and path-based protection. Each of these techniques can be deployed in a shared or a dedicated fashion.

In **link-based protection**, each link has a protection path, and the traffic is switched to the protection path upon the failure of corresponding link. If each working channel on a link has its own dedicated protection wavelength path, it is called **dedicated link protection**. On the contrary, in **shared link protection**, the same wavelength can be used on the common links of two non-disjoint protection paths as long as their corresponding working channels are on different links. Therefore, shared link protection is more capacity efficient than dedicated link protection, and can provide 100 percent recovery from single-link failures. In **path-based protection** schemes, upon the failure of a link, the whole affected working paths are switched to their reserved protection wavelength paths at the end nodes. Therefore, a mechanism is required to inform the end nodes of the affected connections. This makes path protection more complicated than link-based counterparts. Similar to link-based protection, there are dedicated and shared path protection schemes. In **dedicated path protection**, two non-disjoint protection paths must use different wavelengths even if their corresponding working paths are disjoint. Hence, large amount of additional capacity will be required for protection. The advantage of dedicated path protection is that in some cases it is able to protect multi-link failures. Similar to link protection, **shared path protection** can share the same wavelength on the common links of two non-disjoint protection paths, if their corresponding working paths are link-disjoint. Therefore, in terms of capacity it is more efficient than dedicated path protection, while still providing 100 percent recovery from single-link failures.

Different protection schemes have been thoroughly compared in [8] and [21] in terms of speed and efficiency. Unlike protection schemes, restoration methods have been much less considered in the literature. One important note about restoration techniques as mentioned in [22] is that *path restoration has better efficiency, while link restoration has better restoration time.*

### 2.1.2 Ring-based vs. Mesh-based Survivability

Ring-based protection schemes are the basic survivability methods used in optical networks. There are two general types of self-healing rings (SHR); namely the bi-directional line switched ring (BLSR) and unidirectional path-switched rings (UPSR). These two methods are widely considered as generalizations of 1:1 and 1+1 Automatic Protection Switching (APS) respectively. In 1+1 protection, traffic is transmitted simultaneously on two separate fibers from the source to the destination and in a fiber cut, the destination switches over the other fiber and continues to receive data. In 1:1 protection, only one fiber carried the traffic and in a fiber cut, the source and destination both switch over to the protection fiber [5].

In BLSR, nodes that are adjacent to a link failure usually monitor the status of the protection channel. If it is free, the traffic demand will be switched to the protection channel in the reverse direction of the failure. Under the normal operation, BLSR can carry low-priority traffic on the protection bandwidth. Therefore, additional real time signalling is required between the nodes to preempt this low-priority traffic in the event of a failure [5]. In UPSR, traffic is simultaneously transmitted on the working and protection fibers in two reverse directions. The receiver chooses the signal with better quality as the received data. We note that BLSR can be used more efficiently than UPSR, because any two nodes can make similar use of the shared standby capacity around the ring.

The growth of communication networks in response to higher traffic naturally

leads to mesh topologies. Therefore, mesh-based methods emerged for network survivability. One straightforward solution for surviving an overall optical network is to duplicate every transmission path, in the form of rings and protection switching schemes. However, the redundancy costs can be typically very high, compared to a corresponding network which is designed only to serve the working demands. In fact, if the used architecture is not carefully designed, the costs of a survivable network can be twice the cost of a non-survivable network [3]. In mesh-based networks for the same investment in capacity, more working demand can be served in more diverse patterns compared to a corresponding set of rings. In addition, mesh is less costly in long distance networks where bandwidth, size, and geographically diverse path connectivity are highly demanded [3]. However, because of dealing with multiple-path re-routing problems, mesh restoration is not generally as fast as rings.

From economical point of view, mesh restoration schemes are efficient in long-haul networks where cost is more dominated by the total bandwidth-distance product, while ring networks are more profitable in metro networks where cost is mostly determined by terminal equipments [23]. The main advantage of rings is their low cost and high speed when compared to (centrally-controlled) mesh-restoration schemes that require a sophisticated central system with a separate signaling network. This is the reason that despite the need of over 100% redundancy, rings are still preferred in metropolitan areas, where there is less geographic diversity and less required bandwidth than in intercity networks.

The interest in mesh-based survivability techniques has increased progressively, because of their greater flexibility, efficiency, and support for multiple service classes. Moreover, mesh-based methods need less spare capacity for restoration, and can avoid “stranded capacity” effects in rings where one or more ring links are utilized while other links of the ring have valuable but unusable remaining working capacity [3].

Mesh-based networks are also able to organize survivability in response to time-varying patterns of demand. As optical cross-connect and WDM switching technologies evolve, the interest for mesh-based restoration further increased because of the reduced costs of optical-electrical conversion and integration of WDM and electronics. In mesh, spare capacity on one link typically contributes to the “restorability”<sup>1</sup> of many other links. Observing the benefits of ring-based and mesh-based survivability schemes, the trends moved toward having a method which is as survivable and fast as rings but enjoy the flexibility and capacity efficiency of mesh.

## 2.2 $p$ -Cycle Protection Method

The pre-configured protection cycle, known as  $p$ -cycle [10] is an efficient method for designing survivable mesh networks. The basic idea of  $p$ -cycle is to build the protection paths by utilizing the concept of fully pre-cross-connected linear segments [11].  $p$ -Cycle has emerged as an efficient “shared link protection scheme” which benefits from the speed of line-switched self-healing rings while having the capacity efficiency of a mesh-restorable network.  $p$ -Cycle has later been extended to protect nodes and the whole working path in a network. Node encircling  $p$ -cycles [24] are routed through all neighbors of a specific node and protect all the connections traversing through that node.  $p$ -cycle is a proactive survivability scheme with pre-reserved protection paths. The authors of [25] have extended the concept of span-protecting  $p$ -cycles; the main advance in this work is the generalization of the span-protecting  $p$ -cycle concept to protect path segments [26] [27] of contiguous working flow. This effectively extends the span-protecting  $p$ -cycle method to include path protection or protection of any flow segment [26] along a path. More recently, a new technique of failure-independent

---

<sup>1</sup>Restorability is defined in [3] as the fraction of working units that are capable of being recovered by replacement routes through the network.

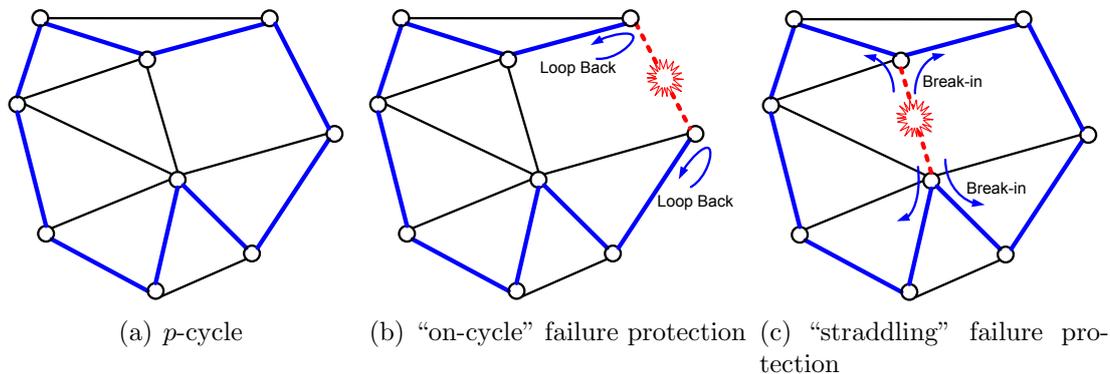


Figure 2.1: Basic operation of  $p$ -cycles for protection of on-cycle and straddling link failures

path-protection (FIPP) has been proposed [12] as a pre-connected, failure independent, path-protecting network architecture. FIPP  $p$ -cycle improves  $p$ -cycles by adding the property of providing end-to-end failure independent path switching against a network component failure while retaining other advantages of  $p$ -cycles. Evidently, this area of research has triggered and attracted recently the attention of several researchers.

### 2.2.1 The Concept of $p$ -Cycle

$p$ -Cycle is based on BLSR protection scheme. The main difference between  $p$ -cycle and conventional rings is that  $p$ -cycle provides two protection paths for each link that straddles the cycle along with the protection of “on-cycle” links. Therefore, unlike the conventional ring protection schemes,  $p$ -cycles can more widely protect the network as a whole [23]. The straddling links can have working capacity but no spare capacity, which is a very unique characteristic of  $p$ -cycle based networks. Figure 2.1 illustrates the basic operation of a  $p$ -cycle for a small network. In this figure, all links of a small network are protected by one  $p$ -cycle. Figure 2.1(b) shows that in case of an on-cycle failure, the end nodes loop back the traffic to the other side of  $p$ -cycle. As illustrated in Figure 2.1(c), when a straddling link fails, there are two alternative protection paths on the cycle.

$p$ -Cycle is a proactive survivability scheme with pre-reserved protection paths. In case of a failure, the only factor that determines *the real time restoration speed* is the time required for the two end-node digital cross-connects (DCC)<sup>2</sup> to do signal bridging and switch to the pre-cross-connected protection path. Each node is aware of required port-to-port connection for each prospective failure [10]. In other words, calculation and connection of protection path is done off-line before the occurrence of any failure.

The switching between working and spare capacity in  $p$ -cycles is functionally similar to rings. It means that in case of any failure only two underlying DCC's end nodes are involved and they only have to perform traffic switching at the end-nodes. However, unlike the BLSR, basically no real-time signaling between end nodes is required to achieve the restoration switching. In rings the working demands and the protection bandwidth are structurally associated. However,  $p$ -cycles are formed only within the spare capacity layer of the network, so the working paths can be freely routed in any desired manner (e.g. shortest paths) like any point-to-point mesh network. Also, a deployed  $p$ -cycle design can be easily shaped and modified by the DCC's, while a ring is basically hardwired in place within the network, once it is deployed. The average length of protection paths (number of links in the path) in a  $p$ -cycle is half that of the corresponding ring for straddling links, and the same as a BLSR ring for on-cycle links.

### 2.2.2 Path-Protecting $p$ -Cycles

The  $p$ -cycle introduced in section 2.2.1 and illustrated in figure 2.1 is generally called “link-protecting”  $p$ -cycle, which can protect links that are on the cycle or directly straddle the cycle. The other type of  $p$ -cycle is “path protecting  $p$ -cycles” which is designed to protect the whole working path. As discussed in 2.1, path-based

---

<sup>2</sup>Digital Cross-Connect (DCC) is *one of the basic components in the SONET infrastructure which is used to manage all the transmission facilities in the central office.* [5]

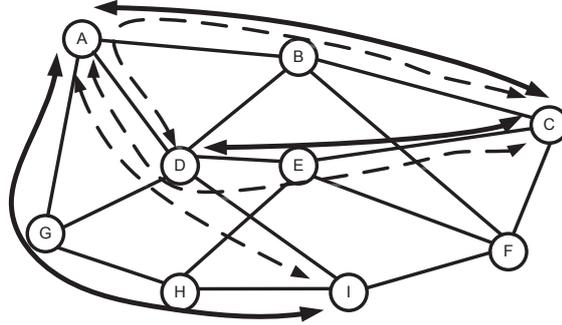


Figure 2.2: A set of three mutually disjoint working routes and their corresponding shared backup paths.

survivability schemes are generally more efficient than their link-based counterparts. However, path-oriented survivability schemes have more design complexity, mainly due to the necessity of addressing “mutual capacity” issue. It means that the spare capacity for each working path should be allocated in a way that it is not blindly used by another working path. One way to address the mutual capacity issue is to only allow working paths that are mutually disjoint to share their protection paths. This is done in shared backup path protection (SBPP) scheme, which is the basis of path protecting  $p$ -cycles. SBPP is a preplanned path restoration scheme which was initially developed in [28] for protection of lightpaths in optical networks. The basic idea of SBPP is that backup routes can share the spare capacity as long as they are disjoint from the working paths. In SBPP, one backup route is predefined for each working path and regardless of what fails on the working path, this predetermined backup route restores the failed demand. Figure 2.2 illustrates a set of mutually disjoint working paths and their corresponding protection paths. The maximum sharing in this example happens on the link AD where three separate working paths are sharing a single unit of spare capacity along the backup route.

It should be guaranteed that in case of a failure in a working path the links and nodes along the corresponding protection path are not affected. Therefore, a protection path should not have any node or link in common with the corresponding working path. It is also not allowed to have any node or link in common with any

other working path which is not mutually disjoint with the corresponding working path. In other words, every working path must be fully disjoint from its own backup route (except at its end nodes), as well as fully disjoint from other working paths that share any spare capacity in their backup paths. Disjointness of working and backup paths has an important advantage called “failure independence”. This means that fault localization is not necessary in real time to determine the restoration response. Fault detection still happens in real time, at the end nodes, but it does not depend on the actual position of the occurred failure. In particular, failure independence is advantageous in transparent or translucent optical networks [5], where fault localization is slow or difficult. This is one advantage of SBPP over failure-specific scheme, such as path restoration [29] or flow  $p$ -cycles [25] that require fault localization.

Although SBPP has very good capacity efficiency, and is end-to-end oriented, it is not actually a protection scheme. It is a preplanned restoration scheme, without backup-path pre-cross-connection property. *The routes of backup paths are decided in advance, but a path must be formed on demand by seizing and cross connecting spare channels on that route when needed. More precisely, SBPP is a failure-independent preplanned path restoration (PR) scheme* [29]. In other words, spare channels for the backup path must be cross connected on the fly upon failure. Therefore, it is not possible to have these channels cross connected in advance of failure. Establishing the required cross connections on the fly is time consuming. Moreover, in order to provide dynamic provisioning, SBPP requires an extensive database to store in every node the global capacity, topology and backup-sharing relationships.

The concept of failure-independent path-protecting (FIPP)  $p$ -cycles was firstly introduced in [30], but the comprehensive evaluation and comparison to other schemes as well as the network design and mathematical model were next discussed in [12]. In summary, it is a relatively simple scheme that extends the  $p$ -cycle concept into a path-oriented version, combining the desirable practical properties. The failure is

not limited to being in a link or path segment immediately adjacent to the end node. One of the main advantages of FIPP is having fully pre-connected protection path that yields the transmission integrity of the backup optical path which is important in order to meet the requirements for speed and optical-path integrity.

FIPP  $p$ -cycle provides protection to the end-to-end primary paths whose end nodes are on the cycle and their routes are all mutually disjoint. The key principle of FIPP  $p$ -cycles is that, similar to SBPP that enforces a disjointness requirement on working routes with shared protection channels, FIPP  $p$ -cycles enforce an a priori disjointness requirement on the end-to-end paths that share any  $p$ -cycle structure. Applying the mutually disjointness constraint to primary paths enables them to share a fully preconnected protection structure, not individual spare channels that still have to be cross connected to form backup paths.

The most important property of FIPP  $p$ -cycles is that their capacity efficiency is similar to SBPP both experimentally and theoretically. Another similarity with SBPP is that FIPP can support completely failure-independent end-node activation and control against either link or node failure. However, unlike SBPP, FIPP  $p$ -cycles do this with fully preconnected protection paths. FIPP  $p$ -cycles enjoy interesting features such as *ring like speeds, minimal realtime signaling, and the assurance of optical signal quality on the protection path when needed* [29]. Moreover, they are able to protect node failures, as well as link failures. Similar to link  $p$ -cycles, in the optimal design of networks with FIPP  $p$ -cycles, most of the cycles are chosen to be in straddling relationship with working path, because it is twice as efficient as fully or partially on-cycle relationships.

Figure 2.3 illustrates a set of five mutually disjoint working paths (“compatible routes”) that are protected by one FIPP  $p$ -cycle. As can be seen, no failure can affect two compatible demands which are protected by the same FIPP  $p$ -cycle. Hence, there is no need for any failure information dissemination. In addition, as long as single

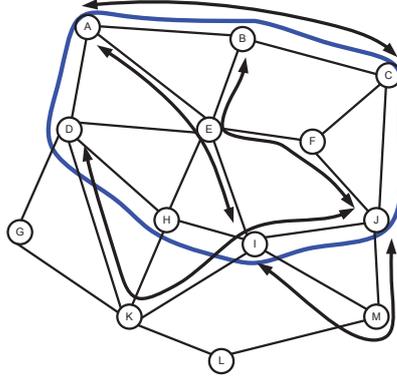


Figure 2.3: A FIPP  $p$ -cycle is protecting a set of five mutually disjoint working paths.

failure scenarios are assumed, there cannot be any contention for spare capacity on a  $p$ -cycle and the mutual-capacity problem is spontaneously addressed.

It may be mistakenly concluded that FIPP solution space is a subset of the SBPP solution space; because FIPP  $p$ -cycle appears to be formed from a specific choice of two backup routes. Therefore, SBPP would be a lower bound for the spare-capacity results of FIPP  $p$ -cycle designs. However, there is one important aspect in which it is not the case. It happens when a path partially straddles the  $p$ -cycle. Unlike SBPP, in FIPP a working path can generally have path segments in common with its own protection structure. In some cases FIPP  $p$ -cycle can outperform SBPP in terms of capacity efficiency. The reason is that the protection path is allowed to have some links in common with the working path. Therefore, the surviving components of the working path can be reused more effectively.

### 2.2.3 Optimal Spare Capacity Design with $p$ -Cycles

The spare capacity design method is usually referred to as the method for determining the amount of spare capacity that must be provisioned in the network to meet the requirement of full survivability of any single failure. Efficient network design using  $p$ -cycles has been extensively investigated in the literature during the past decade [31–35]. The idea of optimal spare capacity design for  $p$ -cycle based restorable networks

was first formulated in [11,36] using integer linear programming (ILP). There are two main design principles for designing a survivable network. One is called “non-joint” or “hierarchical” design problem, where the traffic demands are routed in advance and the required working channels are provisioned without survivability concerns. Then, on the second stage, given the working capacities, a minimum cost allocation of spare capacity on the links is determined so that the disrupted flow can be safely rerouted, in case of a link failure. This second stage problem is referred to as the spare capacity assignment (SCA) problem [37]. Another design principle is to solve the problem jointly which was firstly modeled in [3]. This principle which is called “joint optimization problem” attempts to optimize the choice of working routes in conjunction with the placement of spare capacity to achieve the objective. In each design principle, there can be two scenarios with two different objectives. One design scenario is to achieve the highest level of restorability for a given set of existing spare capacity. The second scenario is the reverse of the first one; that is the minimum set of spare capacity is generated such that 100% restorability is ensured. These two basic scenarios were developed and tested in [36] for non-joint network design based on  $p$ -cycles. The results show that  $p$ -cycles enjoy mesh-like efficiency, although being based on rings. In other words, 100% restorability can be achieved in  $p$ -cycles with little or no additional spare capacity than in a conventional mesh-restorable network [37]. Furthermore, various studies declare that the joint optimization of working path and  $p$ -cycle placement is the most efficient of the fast protection methods [32, 38, 38]. However, this requires much more complex hardware and software equipments. The size of ILP model grows exponentially even when the network is not as dense.

In recent years, there have been some attempts for extending the  $p$ -cycle scheme for protecting multicast traffics [39–42]. For designing survivable multicast networks with  $p$ -cycles, every link of all multicast trees should be protected by  $p$ -cycles with minimum spare capacity. This would result in ILP problems which by far has much

higher number of variables and constraints than the case where only node-pair demands are considered. A set of heuristics to cope with this problem are introduced in [39,40] where ILP based methods are proposed for provisioning static and dynamic multicast sessions. In [41] another method called intelligent  $p$ -cycle is introduced for protecting dynamic multicast sessions and there it is shown that it outperforms the method presented in [40].

In order to achieve the optimal design, *conventional algorithms need to enumerate cycles in the network to form a candidate set, and then use an ILP model to find an optimal set of  $p$ -cycles from the candidate set* [43]. However the number of possible  $p$ -cycles grows exponentially with the number of nodes and links in the network. This makes the problem unsolvable in a reasonable amount of time. It is well known that *the design of a min-cost set of  $p$ -cycles to protect a given set of working flows is an NP-hard problem* [32].

One alternative to deal with this problem is to consider just a limited number of promising cycles and find the optimal solution with the restricted possibilities. However, the optimal solution of the original problem is no longer guaranteed. Several heuristics have been proposed in the literature for preselecting the most promising eligible cycles in large scale networks [23,32–34,36,44,45]. One of the most common algorithms is to sort and pre-select a fraction of promising  $p$ -cycles based on their “A Priori Efficiency” (AE) metric which is measured as the number of protected links, divided by the cost of the  $p$ -cycle [32].

Another alternative to deal with the difficulty for achieving the overall optimal solution of  $p$ -cycle based network design is to decompose the ILP model. One of the most famous and efficient decomposition techniques is “Column Generation” (CG) algorithm. The idea of column generation algorithm is to only generate the variables when needed, i.e., when the reduced cost of a variable is negative [46]. The CG algorithm gives the optimal solution by generating only a fraction of the possible  $p$ -cycles

which are implicitly enumerated. In CG algorithm, the linear programming problem is divided into “master” and “pricing” problems. The master problem contains a restricted set of variables with the main objective subject to some of the original constraints along with some implicit constraints. Master problem gives a relaxed linear programming (LP) solution where the integrality constraints of variables are removed. In order to achieve the optimal integer solution to the master problem, a branch-and-price algorithm is needed with the lower bound given by the optimal LP solution [47]. The objective of the pricing problem is to minimize a so-called “reduced cost” and pass a new promising column to the restricted set in master problem. [47–49]

The first attempt for using CG in the design of  $p$ -cycle based networks was in solving the joint optimization problem [38,50]. In [50], a CG algorithm is implemented to achieve close to optimal solutions for the joint routing and protection design in  $p$ -cycle based networks. Given a network and a set of connection demands, the total capacity of the network is minimized. The initial solution for CG algorithm is a set of shortest paths, one for each demand, and a set of dummy  $p$ -cycles which can only protect one link and therefore are so expensive that they will never occur in the optimal solution. In each iteration of the CG algorithm, the path and the  $p$ -cycle with the minimal reduced cost are found. If there is no path or  $p$ -cycle with a negative reduced cost, it can be said that the optimal solution of master model is found; otherwise, the improving path or  $p$ -cycle is added to the restricted master model and the algorithm iterates. In [51] the authors apply the CG algorithm for the network design based on FIPP  $p$ -cycles. It is shown that using their CG algorithm, the cost can be improved up to 37% compared to the solutions where only restricted promising set of candidate cycles are considered. In [52] and [53] the efficiency of link and path  $p$ -cycles are compared to classical shared link and path protection schemes when CG algorithm is applied for finding the optimal solution in all cases.

More recently,  $p$ -cycle network design without cycle enumeration [43] have also been introduced. Further efforts have been made to explore the efficiency of non-simple cycles whose nodes can be traversed more than once [45, 54]. Clearly, non-simple cycles can yield higher capacity efficiency than simple cycles, especially in network-areas where elementary cycles cannot be deployed. However, this improvement can be obtained at the cost of much higher design complexity, because the number of non-simple candidate cycles exponentially increase at a much higher rate than the simple counterparts. Moreover, it is shown in [34] that the increase in efficiency achieved by deploying non-simple  $p$ -cycles is negligible; they can introduce too much delays for a connection in protection state and the computation time is increased, so it is not recommendable.

#### 2.2.4 Multiple Failure Survivability with $p$ -Cycles

Failure scenarios considered thus far, are “single failures”, and that means single fiber cuts or more generally, cuts of single edges of the network graph. In almost all survivability schemes, the objective is to replace the affected working paths in case of any single network failure. Networks that are fully restorable to single cuts are often called “100% restorable”. However, higher-order failure combinations can make such networks unable to fully recover. In other words, making a network fully restorable to single failures is not a guarantee that the availability of the service in the occurrence of higher order failures will be 100%. Several approaches have therefore been designed to improve the robustness of mesh transport networks against dual-failures. These approaches have either considered (pre-failure) strategies for addition of further protection capacity to achieve full or partial dual-failure survivability [55], [56] or have assumed reconfiguration of protection resources after the occurrence of the first failure to better withstand future failures [57], [58]. One alternative to cope with multiple failures in  $p$ -cycle based networks is  $p$ -cycle reconfiguration

that can be achieved by using static or dynamic (reconfigurable)  $p$ -cycles. Static  $p$ -cycle reconfiguration means that after a first failure the  $p$ -cycles remain as initially configured and the same set of  $p$ -cycles are used for recovering subsequent failures. Conversely, in dynamic  $p$ -cycle reconfiguration the subsequent failures are recovered by finding new  $p$ -cycles in the remaining intact part of the network upon the first failure. Static reconfiguration is useful when dynamic  $p$ -cycle design is not possible or the reconfiguration after a first failure is not completed. These cases are considered in [59] and [60].

Usually the study of multiple failure survivable networks is simplified to considering only dual failures, because occurrence of more than two failures at the same time is very unlikely [9]. The tradeoff between the number of deployed  $p$ -cycles and the survivability of dual fiber duct failures is investigated in [59]. In [60], it is assumed that dual failures are ordered events and the individual failures occur independently, such that the recovery of the first failure is completed before occurrence of the second failure. It should be noted that dual failure scenarios are only considered within one cycle, otherwise multiple failures can be protected by multiple separate  $p$ -cycles. Results in [60] show that network designs with the minimal number of cycles and optimal capacity objectives are only able to restore around a half of the connections after the second failures. In [61], another mechanism called multi failure survivability (MFS) is introduced for recovering multiple failures one at a time. The results indicate that networks with higher average nodal degree are more likely to be survived against multiple failures. Authors in [62] discuss the cases where the second failure occurs before recovering the first failure. Therefore, a fast readjustment of the  $p$ -cycles are required to temporarily protect the vulnerable working paths. The set of  $p$ -cycles can be redeployed either by a global optimization (where the whole network topology is readjusted) or by an incremental optimization (where only the vulnerable demands are re-protected by additional cycles).

In [63], the authors propose a method for dual-failure restoration by dynamically repairing  $p$ -cycles and compare it with incremental and complete dynamic reconfigurations. They studied the additional spare capacity required for dual failure restorability for each method and found that the efficiency of their dynamic repair method is in between complete and incremental reconfiguration schemes. It is clear that complete reconfiguration of  $p$ -cycles after the first failure is the most efficient method. Another article which discusses about  $p$ -cycle reconfiguration is [64] where the demands are divided into different service classes and dual failure survivability is provided to the highest priority demands called platinum traffic.

More recently, the authors of [65] have argued that, in addition to the above mentioned approaches, reductions in the physical repair time of failures can also enhance service availability. They showed that an economic strategy exists for balancing the tradeoffs between capacity investment and Mean Time To Repair (MTTR) reduction efforts to achieving high service availability in networks designed to be 100% restorable against single failures. The authors of [9] studied the availability in span-restorable mesh networks. The availability analysis is based on the computational analysis of the restorability of a network to all possible dual-failure scenarios. In [66], the authors developed an analytical expression for the availability of paths in networks using  $p$ -cycles as the protection mechanism. The model presented is based on the calculation of the unavailability caused by the effects of dual-failures and the authors have used the concept of “cutsets method” or “protection domain” to determine the service availability. An availability-aware service provisioning method in  $p$ -cycle based mesh networks is presented in [2]; therein, the service availability is analytically derived as a function of the span unavailability, using the concept of protection domain. The spare capacity is allocated, through a non-joint optimization model, to meet the availability requirement of the end-to-end traffic. More recently, this availability-aware network design method has been also applied for FIPP [67].

### 2.2.5 Availability Analysis of $p$ -Cycle-Based Networks

Availability of a system is defined as *the probability of the system being found in the operating state at some time  $t$  in the future given that the system started in the operating state at time  $t = 0$*  [3]. The availability can be obtained by dividing MTTF (Mean Time to Failure) to the sum of MTTF and MTTR (Mean Time to Repair). The availability of a service path is influenced by many factors such as *the statistics of network element failures, the statistics of repair times, mean restoration time, etc.*

In [9], it is shown that for the determination of expected service path availability in long-haul networks, the effects of dual link-failures are in fact much more important than other failure scenarios, and considering dual link-failures only is sufficient to obtain a good estimate of the expected availability of service. The analytical expression for the availability of paths in a  $p$ -cycle-based network was firstly introduced in [66]. The model presented is based on the calculation of the unavailability caused by dual-failures.

One of the most common and practical approach for finding service availability in a network is “cutsets method”. In this method, failures that cause service outage are divided into categories. Then the probability of unavailability in different categories are added in order to obtain an estimate of the average service unavailability. To develop the equations for path availability in a  $p$ -cycle protected network, the path is divided into “protection domains”. A path may cross several protection domains between its origin and destination nodes. A path is said to cross a protection domain associated to a  $p$ -cycle, if at some point that path is protected by that  $p$ -cycle. Two slightly different definitions are given for a protection domain in [66] and [2]. In [66], if a link on a path was protected by a  $p$ -cycle as an on-cycle link and another link on the same path was protected as a straddling link, then these two links were counted as two different domains. However, in [2], a “protection domain” is defined as the set of links which are protected by the same  $p$ -cycle. In other words, all links in a path

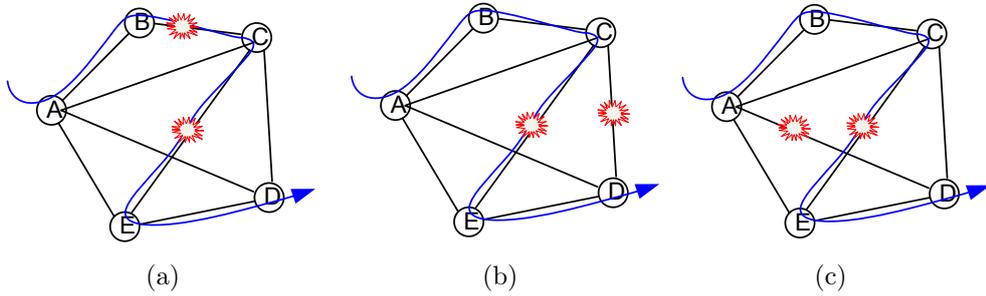


Figure 2.4: Three categories of dual failure scenarios which result in service outage protected by the same  $p$ -cycle belong to the same protection domain and hence the two links in the above case also belong to the same domain.

As discussed in [3] for systems with elements in series, the unavailability of each element needs to be added up, which is an approximation considering that individual unavailabilities are very small, and for systems with elements in parallel the unavailability of each element is multiplied to obtain the total unavailability. Since the protection domains of a path are in series, the unavailability of a path in a  $p$ -cycle protected network can be expressed as the sum of the unavailability of the path in the different protection domains crossed. Therefore the unavailability of each section of the path, which belongs to the same protection domain, must be analyzed.

In [66] and [2], the authors try to derive all possible combinations of dual failures within the protection domain that can result in an outage on the corresponding service path. In [2], six different categories of dual failures which lead to unavailability are considered. Two of these scenarios are for on-cycle links, and the rest are for straddling links in a given protection domain. Each of these sequences is independent from the others, i.e., with respect to a given path a dual link-failure can only belong to one of these sequences. The physical unavailability of each link is assumed to be the same.

Figure 2.4 shows a protection domain according to the definition in [2]. Three categories of dual failure scenarios that result in service outage are illustrated for one path of concern. In part (a) the outage is independent of the order of failure

occurrence, whereas in parts (b) and (c) there will be a service outage for the path of concern only if the first failure occurs on the straddling links which do not belong to the path. In [2], the  $p$ -cycles are assumed to be “fully loaded”, i.e., they *provide restoration to two units of working capacity in all straddling links and one unit of working capacity to all on-cycle links*. Results declare that the size of  $p$ -cycles is very important in determining the availability of service paths traversing their domains [66]. Smaller  $p$ -cycles will allow much higher availability to be offered to paths. However, as we know smaller  $p$ -cycles are generally less capacity efficient than larger ones. Therefore, there is a trade-off between capacity efficiency and availability based on the size of candidate  $p$ -cycles.

After deriving the equation for the overall unavailability of a path, different factors can be investigated for comparing the unavailability of a path in a given domain depending on whether the path is an on-cycle or straddling path for the cycle associated to that domain. Former studies in [57] show that *the amount of spare capacity required to protect all demands against any dual link-failure is typically in the order of three times the amount required to protect against single failures, and the total capacity cost for the whole network would often increase by more than 50%* [66]. Obviously, this is too costly for most network operators. Therefore, in [66] an alternative approach is presented, which consists of improving the availability of only selected service paths instead of trying to improve the availability of all paths. Results in [66] show that the unavailability of both on-cycle and straddling links is proportional to the number of on-cycle links. It is also shown that *the unavailability of the straddling path is 25% lower than that of the on-cycle path for all values of the number of on-cycle links*. The reason of this difference is that on-cycle links have longer protection paths. Therefore, on average they will be more vulnerable to a secondary failure in their protection path compared to straddling links.

Two new models for  $p$ -cycle networks are introduced in [66] for joint optimization of demand routing and spare capacity allocation while taking the priority of service paths into account. The first model is called selectively enhanced availability capacity placement (SEACP) which guarantees that selected priority paths will be routed exclusively on straddling links, therefore enjoying an availability improvement, whereas other paths are routed either on straddling or on-cycle links. Results show that with 3% additional capacity 60-70% of priority demands can be served in this model. The second strategy is called multi restorability capacity placement (MRCP). This strategy offers two protection options to selected priority paths by routing them on straddling links and allowing them to access either sides of the cycle they straddle. Results show that in MRCP, capacity requirement increases rapidly by increasing the priority demands. However, the availability of priority paths with MRCP is expected to be very much higher than that of priority paths with SEACP.

Another approach which is addressed in [2] is to define an upper bound for the maximum value of unavailability of all end-to-end working paths. Afterwards, an optimization model is provided including new constraints for bounding the unavailability of each service path. This optimization model is used to find the minimum cost capacity placement that guarantees protection of every working path against single link failures. This model ensures that the unavailability of all end-to-end working paths is less than a certain user set upper bound. The physical unavailability of each link is assumed to be the same.

The first work for availability analysis of FIPP  $p$ -cycles is given in [67] where the categorization of different dual failure scenarios are extended for different paths of concern. An availability-aware design method for FIPP  $p$ -cycles is also proposed in this work where the network is designed based on availability constraints. Results declare that FIPP  $p$ -cycles *require more network capacity (8-13%) in order to obtain the same level of availability that basic  $p$ -cycle method achieves.*

## 2.3 Survivable Network Design based on $p$ -Trail

The concept of  $p$ -cycle was later generalized in [13], by observing that the high speed protection capability of rings and  $p$ -cycles is not due to their circular topology but rather because their protection routes are pre-cross-connectable. This generalization leads to the definition of pre-cross-connected trails or  $p$ -trails. Similar to  $p$ -cycles,  $p$ -trails achieve the speed of rings with the efficiency of mesh, but they are more flexible than  $p$ -cycles. Theoretically, a  $p$ -trail-based network design can yield a better capacity efficiency compared to the  $p$ -cycle solution, because  $p$ -cycles can be viewed as a special case of  $p$ -trails. The potential advantages of  $p$ -trails over  $p$ -cycles have been explored in [1], where the authors provide an ILP model for  $p$ -trail based network design. In their work, a set of promising non-simple  $p$ -cycles and  $p$ -trails are chosen without enumerating a candidate set in advance. However, the ILP model presented in [1] is not scalable (as the results indicate) which prevents it from obtaining insights on networks of practical sizes. Additionally, the authors do not present a mathematical proof about the overall optimality of their solutions.

The main advantage of the  $p$ -trail over conventional shared link protection schemes is that in a  $p$ -trail all of the protection units are pre-cross-connected along the protection paths; thus, failure recovery can be performed at the “speed of ring”. This feature can be achieved when there is no “branch point” amongst the protection paths. As explained in [13], a node  $X$  in the network is a branch point if *no matter how the protection capacity is pre-cross-connected at  $X$ , there exists a failure scenario for which some needed protection path that has  $X$  as an intermediate node is not properly pre-cross-connected at  $X$* . In order to avoid branch points, the protection plan of the network is provisioned such that for every node  $v$ , if  $e_1$ ,  $e_2$ , and  $e_3$  are three distinct edges that each have  $v$  as an end node, and the protection path of some demand contains both  $e_1$  and  $e_2$ , then no protection path of any demand contains both  $e_1$  and  $e_3$  [13]. This point is illustrated in Figure 2.5 where two working units on links

$AB$  and  $CD$  are required to be protected. The protection scheme in Figure 2.5(a) incur a branch point at node  $E$ . This means that, if the two protection paths share a spare channel on link  $CE$ , the cross-connection at node  $E$  can only be determined *after the occurrence of the failure*. Accordingly, if the protection paths are needed to be pre-cross-connected, there should be two distinct spare units on link  $CE$ ; i.e., one unit for each of the two protection paths traversing through this link. On the contrary, the protection scheme in Figure 2.5(b) removes the branch point and therefore the spare units on links  $AE$  and  $CE$  can be shared between the two protection paths while maintaining pre-cross-connection on all the intermediate nodes.

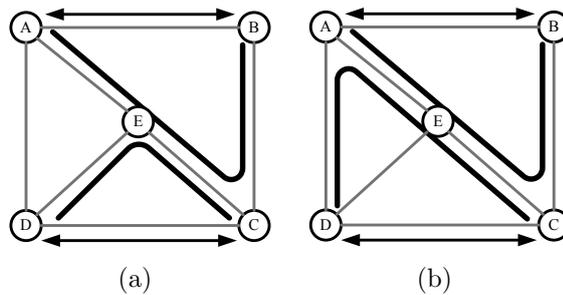


Figure 2.5: Protection schemes with and without branch points

## 2.4 Optical Access Networks

During the past decade, migration from a copper-based plant to a fiber-based plant has been expedited in the telecommunication infrastructure, starting with the wide area networks (WANs) that provide connectivity between cities and progressing through the metropolitan area networks (MANs) that provide connectivity between service provider locations within a metropolitan area. Meanwhile, local area networks (LANs) that interconnect nodes within an individual location have seen average bit rate transitions from 10 Mbps to 1 Gbps over copper cabling [68]. While significant bandwidth improvements occurred in service provider networks (i.e., WANs and

MANs), as well as at the subscriber premises (i.e., LANs), the same level of advancement has not been observed in access networks that provide the link between the private customer networks and the public service provider networks. Although the existing broadband solutions, i.e., digital subscriber line (DSL) and cable modem (CM) networks have made marginal improvements in bandwidth capacity, they are unable to keep up with the increasing bandwidth request of emerging services.

To address the capacity challenges for now and the foreseeable future, fiber to the home/premises (FTTH/FTTP) has emerged for providing various services and applications in “last mile”<sup>3</sup> infrastructures, such as triple play, video on demand, video conferencing, peer-to-peer (P2P) audio/video file sharing, Internet Protocol television (IPTV), multimedia/multiparty online gaming, telemedicine, telecommuting and surveillance [70]. FTTH and more generally FTTX enjoy unique properties such as low loss and extremely wide inherent bandwidth, making it the ideal candidate for providing the required bandwidth to customers for many years to come [71].

To provide a cost efficient and flexible fiber infrastructure in the access network, a passive optical network (PON) can be deployed between service providers and customer premises. In a PON, a shared fiber medium is created by using a passive optical splitter/combiner in the physical plant. Sharing the fiber medium means reduced cost in the physical fiber deployment, and using passive components in the physical plant means reduced recurring costs by not maintaining remote facilities with power. These reduced costs make the PON an attractive choice for access networks, which are inherently very cost sensitive [68]. While providing interesting benefits, the shared medium infrastructure of PONs requires intelligent allocation and management of common resources. As discussed in Section 1.2.2, in the downstream direction from OLT to ONUs, the PON is a point-to-multipoint medium. Conversely,

---

<sup>3</sup>The first (last) mile, also referred to as the subscriber access network or the local loop, is the network infrastructure at the neighborhood level that connects service provider central offices to business and residential subscribers. [69]

in the upstream direction it is a multipoint-to-point network, where ONUs share the same feeder fiber but the upstream optical signal is not received by the ONUs. Since ONUs share the same fiber, their transmissions can collide; hence, contention resolution must be performed to avoid collisions in the upstream direction. Time division multiple access (TDMA) or wavelength division multiple access (WDMA) techniques can be deployed to multiplex in a collision-free way the traffic streams generated by the ONUs onto the common feeder fiber [71].

### **2.4.1 Various PON Architectures and Standards**

During the past decade, various standards have been developed for realizing PONs. The International Telecommunication Union (ITU) has introduced the asynchronous transfer mode (ATM) PON (APON) [72], G.983 broadband PON (BPON) [73], and G.984 gigabit-capable PON (GPON) [74]. On the other hand, IEEE has developed two standards for carrying traffic in the form of Ethernet frames. One is the IEEE 802.3ah Ethernet PON (EPON), also referred to as 1G-EPON standard; the other is 10G-EPON standard developed by IEEE 802.3av task force.

#### **1G-EPON**

Given that 90 percent of data traffic is in the form of Ethernet frames, it is desirable to have an Ethernet based PON structure in order to reduce the adaptation required for exchanging data between LAN and the access network [68]. In addition, EPON can avoid inefficiency and processing delay of segmentation and reassembly, which is induced by fixed data unit size requirement of ATM cells in APONs [68].

EPON is based on time division multiplexing to avoid collision in the upstream transmission. The data rate is 1 Gbps in both downstream and upstream directions; but the line rates increase to 1.25 Gbps due to a 25% bit-to-baud overhead incurred by the 8B/10B line encoding. In order to improve bit error rate and compensate for

optical power attenuation, the 1G-EPON standard specifies the Reed-Solomon code (255, 239) as optional forward error correction (FEC) with an electrical gain of 5.9 dB.

Generally, the bandwidth allocation procedure is carried out at the OLT, using a medium access control protocol. In the IEEE 802.3ah EPON (1G-EPON) standard, the so-called multi-point control protocol (MPCP) [75] is implemented at the MAC layer for exchanging necessary control information between the OLT and ONUs. MPCP is responsible to perform bandwidth allocation, auto-discovery, and ranging. In MPCP, a GATE message is used by the OLT to convey information to the ONU about the size of the allocated transmission window and the schedule of its transmission and a REPORT message is used by the ONU to transmit information to the OLT about its queue occupancies.

## **10G-EPON**

EPON has been recently extended to 10G-EPON in order to provide a ten-fold data rate of 10 Gbps. 10G-EPON has emerged as a promising candidate for next-generation high data rate access systems [17, 76]. This new PON has been standardized under the IEEE 802.3av task force with the aim of developing the physical layer specification and management parameters. The 10G-EPON standard provides symmetric 10 Gbps downstream and upstream, as well as asymmetric 10 Gbps downstream and 1 Gbps upstream data rates. In order to provide backward compatibility with the existing and widely deployed 1G-EPON, the OLT in a 10G-EPON is equipped with dual-rate receivers for receiving data from 1G and 10G-ONUs. Furthermore, the downstream transmission channels are separated for sending downstream data and control traffic to 1G- and 10G-ONUs.

The IEEE 802.3av task force focused only on the physical layer and divided it into

four sublayers, namely, the reconciliation sublayer, symmetric and asymmetric physical coding sublayer, physical medium attachment, and physical medium dependent sublayers for symmetric 10 Gbps data rates and asymmetric 10 Gbps downstream and 1 Gbps upstream data rates, while maintaining complete backward compatibility with 1 Gbps EPON equipment. Therefore, the MAC protocol of 1G-EPON remains unchanged. In fact, the 10G-EPON MAC protocol is an extension of MPCP for 1G-EPON that includes enhancements for management of 10G-EPON FEC and inter-burst overhead.

The 64B/66B line coding in 10G-EPON reduces the bit-to-baud overhead to 3%, compared to the 25% overhead in 1G-EPON, which is incurred by 8B/10B line encoding. The burst signal format of 10G-EPON is similar to that of 1G-EPON, except that the receiver settling time of 10G-EPON is twice of that in 1G-EPON, i.e., 800 ns in 10G and 400 ns in 1G-EPON. The laser on/off time and clock data recovery time of both standards are the same (512 ns and 400 ns, respectively) [17].

The wavebands utilized for upstream (US) and downstream (DS) transmissions of 1G- and 10G-EPON standards are illustrated in Figure 2.6. As can be seen in Figure 2.6(a), 1G-EPON allocates a 100 nm waveband centered at 1310 nm for upstream transmission and a 20 nm window centered at 1490 nm for downstream transmission. As shown in Figure 2.6(b), the downstream wavelength of 10G-EPON is allocated in a window between 1575 and 1580 nm (with a typical value of 1577 nm), which is outside of the analog RF video distribution band. Conversely, the upstream wavelength of 10G-EPON is allocated in a 20 nm window centered at 1270 nm which is completely covered by a part of the 1G-EPON upstream waveband.

Unlike 1G-EPON, implementation of an FEC code is mandatory in 10G-EPON in order to realize its new power budget class. Considering several aspects of various FEC codes, such as the gain, circuit size, and latency associated with encoding and decoding, the Reed-Solomon code (255, 223), which is a linear cyclic block code, was

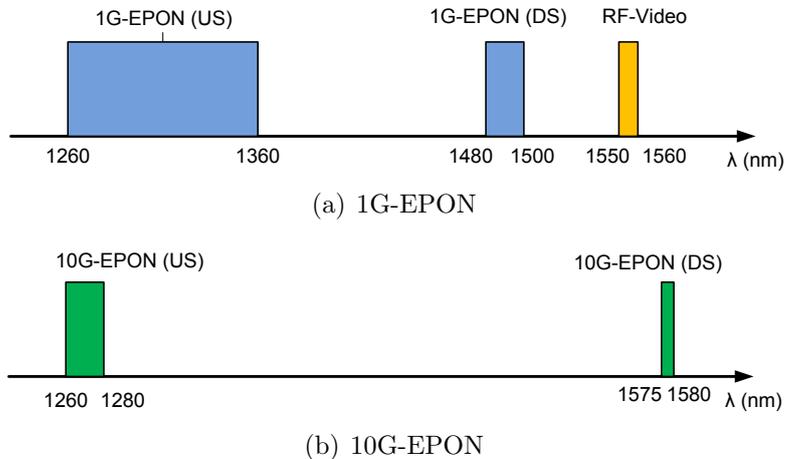


Figure 2.6: Waveband allocation in 1G- and 10G-EPON

adopted for 10G-EPON to enhance the FEC gain and alleviate optical transceiver specifications. This FEC code has an electrical gain of 7.2dB, which allows lower power signal detection with the same bit error rate compared to the optional FEC utilized in legacy 1G-EPONs. However, the FEC in 10G-EPON results in up to 12.9% overhead which is the major overhead component in 10G-EPON [77].

The backward compatibility requirement of the new and existing EPONs introduces several technical challenges and difficulties on the specification work such as a high power budget exceeding 30 dB for symmetric 10 Gb/s transmission, conflicts in wavelength allocation, and dual-rate burst-mode operation at the OLT receiver [78]. Two main techniques are employed for achieving the coexistence of 10G-EPON with 1G-EPON (and analog RF video distribution) systems: WDM overlay in the downstream direction and a dual-rate burst-mode receiver in the upstream direction to support a dual-speed TDM. A typical architecture for realizing the coexistence of 1G- and 10G-EPON is illustrated in Figure 2.7.

In the downstream direction, since the wavebands are distinct, a WDM-overlay is a straightforward way to provide coexistence with the legacy 1G-EPON. On the contrary, as depicted in Figure 2.6, the upstream waveband of 10G-EPON is in fact a subset of the 1G-EPON waveband; hence, a dual-rate burst-mode operation is



and the Stanford University access (SUCCESS) network [83] are some of the proposed WDM metro-access architectures.

The required enabling technologies for realizing future WDM PON and WDM/TDM PON systems have been reviewed in [84]. In WDM PONs, the OLT and ONUs must be capable of sending and receiving data on multiple wavelengths. One straightforward approach is the evolutionary upgraded PON where the OLT is equipped with an array of fixed-tuned receivers and fixed-tuned transmitters for receiving from and sending out data to the ONUs. Accordingly, each WDM ONU supports a subset of more than one wavelength for transmitting and receiving traffic, respectively. A more cost-effective technology for realizing WDM ONUs is to utilize so-called “colorless ONUs” which are wavelength-independent. A colorless ONU makes use of a reflective semiconductor optical amplifier (RSOA) for remote modulation of the upstream data [85]. In this approach, the OLT is equipped with laser diodes to send optical continuous wave (CW) signals to the attached reflective ONUs, where the CW signal is modulated and sent back to the OLT; hence, no light source is required at the ONU. In this approach, it is important to appropriately manage the interference caused by backreflection of upstream signals [86]. If the round-trip transmission is carried out on a single fiber, the signal-to-noise ratio (SNR) is degraded by the interference intensity noise caused by backreflection in the access fibers, and this issue needs to be carefully addressed when colorless ONUs are deployed.

## **2.4.2 Scheduling and Bandwidth Allocation in PONs**

As explained earlier, a PON is a point-to-multipoint medium in the downstream direction and a multipoint-to-point medium in the upstream direction. In other words, only the OLT is connected to all ONUs, and the ONUs can only communicate with the OLT and not with each other. Due to this connectivity pattern, a PON has to utilize a centralized “polling-based” MAC protocol located at the OLT, whereby the

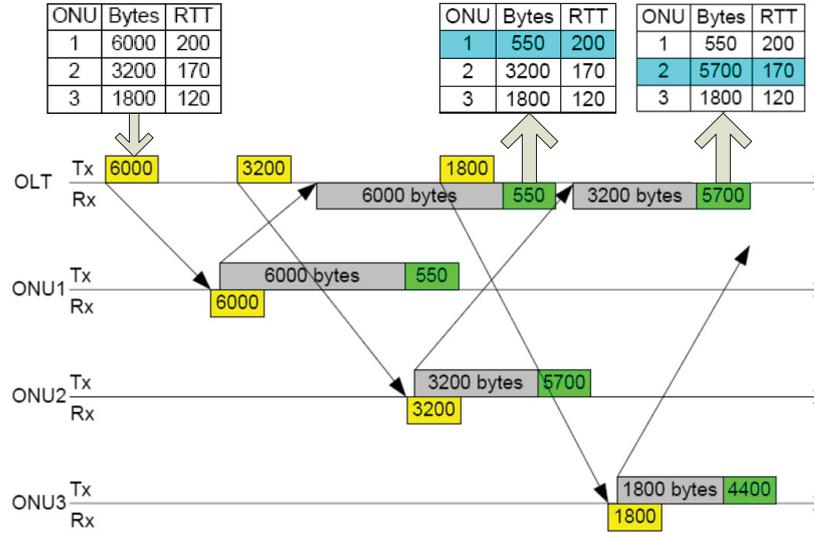


Figure 2.8: Timing diagram of IPACT

OLT polls ONUs and arbitrates their access to the shared PON medium [68]. Given the burstiness of traffic demands in PONs, resource management and allocation is carried out by the process of dynamic bandwidth allocation (DBA). Generally, a DBA process is a cyclic interleaved polling system where the ONUs are polled in turn. The duration between successive polling of each ONU is called the polling cycle. A basic DBA algorithm is the interleaved polling with adaptive cycle time (IPACT), which was presented in [87] for single channel EPON and extended in [88] for multichannel networks. Other variants of IPACT for WDM PON have been investigated in [16] and [89]. In IPACT method, the OLT arbitrates the ONUs in a way that the successive upstream transmissions on the upstream channel are separated in time by only a guard time interval rather than the round-trip time (RTT). Figure 2.8 illustrates a scheduling example for EPON based on IPACT method. In this example, the OLT has to schedule the upstream transmission of three ONUs. The bandwidth request of each ONU and the corresponding RTT is given in a table. Note that the requested bandwidth is updated after receiving each ONU's REPORT message. Based on the RTT information, the OLT can arbitrate the ONU transmissions with a guard time interval between adjacent transmissions.

The DBA problem consists of two subproblems: grant sizing and grant scheduling. Grant sizing determines the length of the transmission window assigned to an ONU for each polling cycle, while grant scheduling indicates the order of ONU grants during a given cycle [68]. Several grant sizing and scheduling methods have been introduced in the literature. The two most frequent grant sizing techniques are “gated service” and “limited service.” In gated service, the grant size for an ONU is simply the queue size reported by that ONU. This scheme provides low average delay, but does not provide adequate control to ensure fair access among ONUs. On the contrary, the limited grant-sizing technique sets the grant size to the reported queue size up to a maximum value for each ONU. Simulation results have shown that there is no average packet delay difference between gated and limited grant sizing [87]. However, limited service sizing scheme prevents any ONU from monopolizing the shared link.

The two subproblems of grant sizing and scheduling are often considered separately yielding to non-joint scheduling and bandwidth allocation methods, where the allocated bandwidth is determined in advance through a grant sizing technique such as limited service or gated service. On the contrary, in joint scheduling and bandwidth allocation methods, the OLT simultaneously allocates transmission grants to ONUs and schedules their transmissions. Sufficient investigation of joint methods is lacking in previous studies. This interesting design scenario will be addressed in Chapters 5 and 6 for next generation WDM PON and 10G-EPON.

In WDM PONs, the scheduling sub-problem can in turn be considered in two layers; a scheduling framework and a scheduling policy. The scheduling framework determines when the OLT makes scheduling decisions, and the scheduling policy is a method for the OLT to produce a schedule. Typically, the choice of the scheduling framework has a large impact on the average queuing delay and achievable channel utilization. A straightforward method is *online* scheduling, where the OLT schedules the transmissions after receiving an ONU’s REPORT and does not wait for the

REPORT messages from other ONUs. Alternatively, in *offline* scheduling, the OLT makes the schedule after receiving bandwidth requests from all ONUs [90]. McGarry *et al.* [91] considered the problem of efficient grant scheduling in a multichannel EPON network. They presented an efficient grant scheduling method for a multichannel EPON based on the so-called Just-in-Time (JIT) scheduling framework. JIT is an online scheduling framework that defines a scheduling pool where ONUs are added to this pool and those in the pool are scheduled as soon as a wavelength becomes available. Another grant scheduling approach was proposed in [92], where under-loaded ONUs are immediately scheduled using the first available wavelength while highly loaded ONUs are deferred until the arrival of all REPORT messages.

Various scheduling policies have been considered for PONs. A simple online scheduling policy for WDM PONs is the next available supported channel (NASC), where the OLT schedules the upstream transmission of an ONU on the earliest available wavelength channel supported by the ONU. The offline grant scheduling problem is solved in [90] for the evolutionary upgrade of WDM PONs, where an evolving number of transmission channels are available. The authors of [90] presented an ILP model based on the scheduling theory, where each ONU is considered as a job, its grant size defines the processing time, and the channels used for transmission on the PON represent machines. Therefore, the problem reduces to a “Parallel Machine” (PM) scheduling problem, where a set of jobs, with specific processing times, are executed on a set of machines [93]. They showed that the least flexible job (LFJ) first with shortest processing time (SPT) first dispatching rule is a good heuristic for this model.

In single-channel PONs, REPORT messages arrive one at a time at the OLT and thus the scheduling is greatly simplified. In a multichannel optical access network, however, multiple REPORT messages may be received concurrently. Therefore, more sophisticated grant scheduling mechanisms are required for handling the bandwidth

demands of multiple ONUs so that efficient bandwidth utilization is achieved. In [94], a control plane was presented for next-generation multichannel access networks, which allows for a flexible upstream wavelength allocation. This control plane operates at two time-scales; one is the microscopic time-scale, which deals with traditional packet access, the other one is the macroscopic time-scale, which assigns connections to upstream optical channels with the objective of optimizing network utilization. In [95], the grant scheduling problem for the so-called STARGATE multichannel EPON (SG-EPON) [96] was formulated as an open shop scheduling problem. The authors of [95] presented a Tabu search heuristic for solving their model using dispatching rules. Their results show that substantial improvements can be achieved in terms of channel utilization and queuing delay when appropriate decisions are made for grant scheduling and channel assignment. More recently, the authors of [97] presented some efficient online scheduling frameworks for WDM PONs with the aim of reducing the idle gaps on each channel. They focused on online scheduling of colorless ONUs, whereby the grant sizes are determined in advance based on a simple gated approach.

## Chapter 3

# Availability-Aware Service

# Provisioning in $p$ -Cycle-Based Networks

In this chapter, we develop a more elaborate model for availability analysis of  $p$ -cycle based mesh networks. We first highlight some shortcomings which make the model and analysis reported in [2] inaccurate and hence we derive a more accurate model, termed as the ApC model, after addressing those subtle issues. Namely, we exhaustively enumerate all dual-failure scenarios which may lead to a service outage on the path through which the demand is routed. We then show that a very meticulous analysis must be done on each protection domain traversed by the service path so that an overestimation of the unavailability of the service is avoided. The resulting ApC model is hence a more accurate model, but with less scalability. We accordingly propose several techniques to address the scalability issues of the ApC model, which results in a smaller overestimation than in [2]. Results show that, in spite of not being able to solve optimally the ApC model, we can get very good estimation of the network unavailability.

The rest of this chapter is structured as follows. In Section 3.1 we present an elaborate analysis of the unavailability in a  $p$ -cycle-based network. Section 3.2 presents some critical issues which must be considered and had been overlooked in previous work [2]. In Section 3.3, we construct an improved availability-aware model that addresses these critical issues. In Section 3.4, we discuss how to solve the ApC model in reasonable amount of time using various ILP heuristic techniques. In Section 3.5, we present the experimental results. Finally, section 3.6 concludes this chapter.

### 3.1 Availability Analysis of $p$ -Cycle Based Networks

The service availability is defined as *the probability of the system being found in the operating state at some time  $t$  in the future given that the system started in the operating state at time  $t = 0$*  [3]. The availability of a service path is influenced by many factors such as the statistics of network element failures, repair time, mean restoration time, etc. One of the most common and practical approaches for finding service availability in a network is the “cutsets method”. In this method, failures that cause service outage are divided into non-overlapping categories and a dual-failure can only belong to one of these categories [2]. This analysis assumes that each span has the same physical unavailability ( $U$ ). In [2], a “protection domain” is defined as the set of spans (on-cycle/straddling) which are protected by the same  $p$ -cycle. The authors of [2], assumed that the  $p$ -cycles are “fully loaded”, which means they *provide restoration to two units of working capacity in all straddling spans and one unit of working capacity to all on-cycle spans*. Accordingly, they partitioned a  $p$ -cycle ( $p$ ) which is protecting the spans traversed by a demand (routed along a path  $r$ ) into four mutually exclusive subsets, as follows:

$O_r^p$  Set of on-cycle spans in  $p$ -cycle ( $p$ ) that are on the working path ( $r$ ).

$O_r^p$  Set of on-cycle spans in  $p$ -cycle ( $p$ ) that are not on the working path ( $r$ ).

$S_r^p$  Set of straddling spans in  $p$ -cycle ( $p$ ) that are on the working path ( $r$ ).

$S_r^p$  Set of straddling spans in  $p$ -cycle ( $p$ ) which are not on the working path ( $r$ ).

Hence, dual-failure scenarios that may cause service outage in a protection domain are classified as follows:

**C-1.** One of the failed spans belongs to  $O_r^p$  and the other one belongs to  $O_r^p$ .

**C-2.** One of the failed spans belongs to  $O_r^p$  and the other one belongs to  $S_r^p$ .

**C-3.** One of the failed spans belongs to  $O_r^p$  and the other one belongs to  $S_r^p$ .

**C-4.** One of the failed spans belong to  $S_r^p$  and the other one belongs to  $O_r^p$ .

**C-5.** Both failed spans belong to  $S_r^p$ .

**C-6.** One of the failed spans belongs to  $S_r^p$  and the other one belongs to  $S_r^p$ .

The unavailability contribution of each category can be achieved by expressions (3.1)-(3.6). We present the analysis of service outage probability for categories **C-2** and **C-4**, the rest is similar and can be found in [2]. In **C-2**, one span failure ( $s_1$ ) belongs to  $O_r^p$  and the other span failure ( $s_2$ ) belongs to  $S_r^p$ . Here, the order in which the failures occur is important. There will be a service outage if a failure occurs first on  $s_2$  (straddling span) and assuming that  $s_2$  is “fully loaded”. Since a straddling span may fail first with a probability of 50%, we can denote the unavailability due to a dual-failure in this category as the expression in (3.2). Notice that for category **C-4**, if the first failure occurred on a span belonging to  $S_r^p$  followed by another failure on a span in  $O_r^p$ , then only half of the traffic will experience service outage. Alternatively, if the first failure occurred on a span in  $O_r^p$  and the second occurred on a span in  $S_r^p$ , then all the traffic of the demand of concern will be disrupted. Given that each

case occurs with a half probability, the resulting unavailability in this category is as shown in equation (3.4).

$$U_{C-1} = |O_r^p| \cdot |O_{\bar{r}}^p| \cdot U^2 \quad (3.1)$$

$$U_{C-2} = \frac{1}{2} |O_r^p| \cdot |S_{\bar{r}}^p| \cdot U^2 \quad (3.2)$$

$$U_{C-3} = |O_r^p| \cdot |S_r^p| \cdot U^2 \quad (3.3)$$

$$U_{C-4} = \frac{3}{4} |S_r^p| \cdot |O_{\bar{r}}^p| \cdot U^2 \quad (3.4)$$

$$U_{C-5} = \frac{1}{2} |S_r^p| \cdot (|S_r^p| - 1) \cdot U^2 \quad (3.5)$$

$$U_{C-6} = \frac{1}{2} |S_r^p| \cdot |S_{\bar{r}}^p| \cdot U^2 \quad (3.6)$$

Since the categories are mutually exclusive, the service outage probability in each domain is  $U_{domain} = \sum_{i=1}^6 U_{C-i}$ . When the protection domains traversed by a route  $r$  are in series [3], the service unavailability can be approximated as follows:

$$U_{path} \cong \sum_{domains} U_{domain} \quad (3.7)$$

The inaccuracy in (3.7) arises from the fact that higher degree polynomials of  $U_{domain}$  have been neglected. Such an approximation is reasonable, as the service unavailability is usually very small and higher degree terms are negligible.

## 3.2 Improved Availability Modeling

We discuss in this section the improvements over [2] and their impact on the unavailability evaluation. Namely, we address two subtle issues which have to be considered to achieve a more accurate availability analysis.

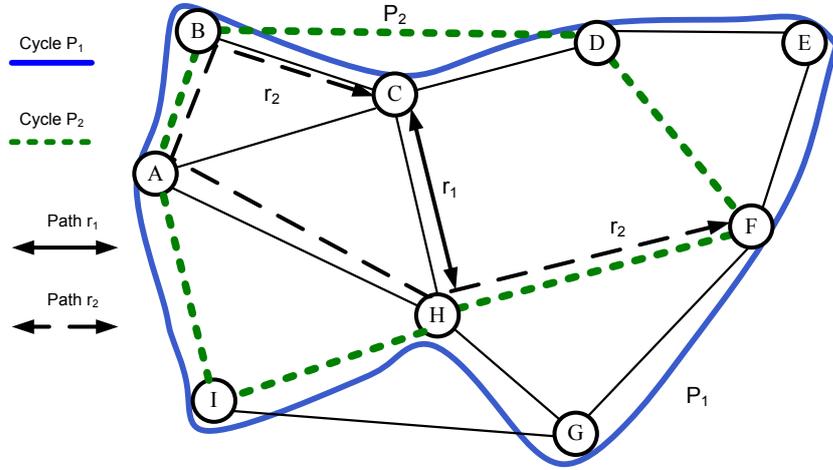


Figure 3.1: Spans traverse two protection domains

### 3.2.1 Overcounting of Spans in the Protection Domain

The first issue concerns the definition of “protection domain” which has to be carefully considered in order to avoid over-estimating the probability of service outage. It should be noted that one span can traverse several protection domains. However, the failure in a span only affects the availability of its corresponding protection domain. In other words, when dealing with dual-failure scenarios, only the cycles should be considered which really protect the working channels on a failed span. We illustrate this in Figure 3.1; span  $C - H$  is fully loaded by a two-unit working route  $r_1$ , which is protected by  $p$ -cycle  $p_1$  ( $A - B - C - D - E - F - G - H - I - A$ ). Although, both working channels on the straddling span  $C - H$  are protected in the same protection domain, it clearly straddles another cycle  $p_2$  ( $A - B - C - D - F - H - I - A$ ), as shown. For deriving the service unavailability, it should be noted that a failure on span  $C - H$  only affects the availability of protection domain  $p_1$  even though it straddles another cycle  $p_2$ .

For working path  $r_2$  between  $C$  and  $F$  and going through links  $C - B$ ,  $B - A$ ,  $A - H$ ,  $H - F$ , we assume that spans  $A - H$  and  $B - C$  are protected by  $p_1$  and the rest of the spans in this path are protected by  $p_2$ . As a result, the on-cycle span  $A - B$  should

only be considered in protection domain  $p_2$  not  $p_1$ . Furthermore, although span  $F - H$  straddles  $p_1$ , it should not be considered in this protection domain, because it is being protected by  $p_2$  as an on-cycle span. Considering protection domain  $p_1$  and path  $r_2$ , table 3.1 shows the spans belonging to different availability sets in our model and compares them with the model of [2]. The effect of resolving the overestimation is obvious in the last row, which shows the unavailability of working path  $r_2$  in protection domain  $p_1$ .

Table 3.1: Comparing the Spans Belonging to Different Availability Sets in Our Model with the Model of [2]

	<b>New definition</b>	<b>Prior work [2]</b>
$O_{r_2}^{p_1}$	$B - C$	$A - B, B - C$
$O_{\bar{r}_2}^{p_1}$	$A - B, C - D, D - E$	$C - D, D - E, E - F$
	$E - F, F - G, G - H$	$F - G, G - H, H - I$
	$H - I, I - A$	$I - A$
$S_{r_2}^{p_1}$	$A - H$	$A - H, H - F$
$S_{\bar{r}_2}^{p_1}$	$C - H$	$A - C, C - H, B - D$
		$D - F, G - I$
$U_{r_2}^{p_1}$	$16U^2$	$39.5U^2$

This delicate issue needs to be considered in the availability analysis in order to avoid over counting and over estimation of service unavailability. Therefore, we redefine the subsets  $O_r^p$ ,  $O_{\bar{r}}^p$ ,  $S_r^p$ , and  $S_{\bar{r}}^p$  as follows:

$O_r^p$  Set of on-cycle spans in  $p$ -cycle ( $p$ ) that are on the working path ( $r$ ) and protected by ( $p$ ).

$O_{\bar{r}}^p$  Set of on-cycle spans in  $p$ -cycle ( $p$ ) that are not on the path ( $r$ ) and also those on-cycle spans traversed by path ( $r$ ) but not protected by  $p$ -cycle ( $p$ ).

$S_r^p$  Set of straddling spans in  $p$ -cycle ( $p$ ) that are on the working path ( $r$ ) and protected by ( $p$ ).

$S_{\bar{r}}^p$  Set of straddling spans in  $p$ -cycle ( $p$ ) that are protected by ( $p$ ) excluding those in  $S_{\bar{r}}^p$ .

Note that  $O^p = O_r^p \cup O_{\bar{r}}^p$  is the set of on-cycle spans, and  $S^p = S_r^p \cup S_{\bar{r}}^p$  is the set of straddling spans which are protected by  $(p)$ . We also note that  $S_{\bar{r}}^p$  is the subset of straddling spans which are either on  $r$  but protected by  $p$  as part of another demand  $r' \neq r$ , or not on route  $r$  (again protected by  $p$  as part of another route  $r' \neq r$ ). In section 3.3, we investigate how the ILP model needs to be modified in order to take into account the new definition of subsets. As we will see, it requires introducing a new set of variables, and therefore impact the scalability of the solution of the ILP model.

### 3.2.2 Exhaustive Enumeration of the Dual Failure Scenarios

Another issue in prior work [2] is that the six dual span-failure scenarios do not cover all cases that lead to service outage in one protection domain. There is one additional scenario that causes outage and has been overlooked. This new scenario consists of dual-failures on two on-cycle spans which both belong to the path of concern  $(r)$ . This scenario, which is hereafter called category 7, **C-7**, causes an outage regardless of the order and location of the failures. The number of combination of dual-failures in **C-7** is given by  $\binom{|O_r^p|}{2}$ ; hence the unavailability contribution can be written as:

$$U_{C-7} = \frac{1}{2}|O_r^p| \cdot (|O_r^p| - 1) \cdot U^2 \quad (3.8)$$

In the following, we illustrate this new category with two examples. Figure 3.2 shows a path of concern consisting of two on-cycle spans and one straddling span, all being protected in one protection domain. Upon the occurrence of the first failure on  $B - C$ , the end nodes switch the traffic to the backup path  $B - A - F - E - D - C$ . A second failure on span  $F - E$ , affects the backup path of the service restored from the previous failure. Meanwhile, the second failure can not be recovered, because the first has already affected its backup path. Therefore, this dual failure scenario results in service outage regardless of the order of failures. Figure 3.3 shows that a

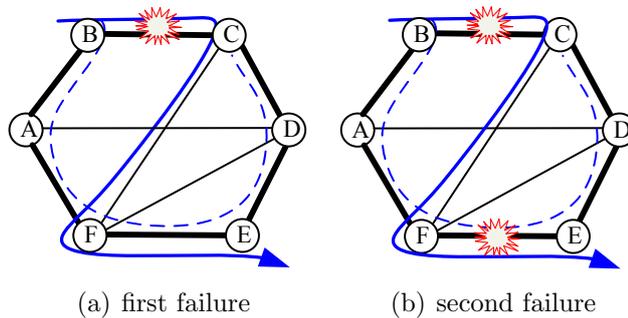


Figure 3.2: New category of dual failure scenarios which result in service outage

dual-failure involving two on-cycle spans from the path of concern will cause service outage even if both spans are adjacent. The first failure on  $B - C$  invokes the end nodes to switch the traffic to the backup  $B - A - F - E - D - C$ . At first glance, it seems that when the second failure takes place on  $A - B$ , the connection can be successfully recovered through the backup  $A - F - E - D - C$ . However, this is not true, because when the cross connect switch at  $A$  attempts to loopback the traffic, it finds that the backup path is already in use due to the switching after the first failure. The switch would be able to do this job if the backup path was preempted after the occurrence of the second failure. In other words, it requires “stub release” capability<sup>1</sup> and implies signalling on the switches which is not desired [3, 99].

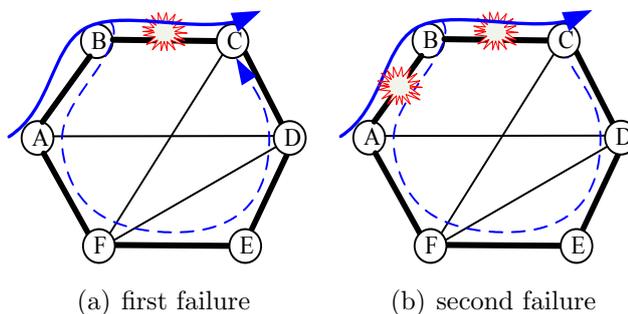


Figure 3.3: Dual failure in two adjacent on-cycle spans

Note that the list of categories we presented for dual-failure scenarios which result

<sup>1</sup>“Stub release” capability means the ability to *release the surviving upstream and downstream portions of a failed working path and make the freed capacity available to the recovery process* [3]. This is equivalent to “Hard Label Switched Path (LSP) Restoration” which is defined in [98]

in service outage is exhaustive. The three other remaining scenarios: (i) two spans in  $O_r^p$ , (ii) two spans in  $S_r^p$  and (iii) one span in  $O_r^p$  and one span in  $S_r^p$  do not contribute to the unavailability of the service path  $r$ . Accordingly, the service outage probability in each domain can now be obtained as follows:

$$U_{\text{domain}} = \sum_{i=1}^7 U_{C-i}. \quad (3.9)$$

### 3.3 The ApC Model

We propose an improved ILP model, called ApC model. The model optimizes the allocation of spare capacity in order to find the minimal cost capacity placement that allows us not only to guarantee that every unit demand is protected against single span failures but to also ensure that the availability of any service path is not less than a desired minimum value. The routing of the demands is done using a standard shortest path algorithm and ahead of the placement of the  $p$ -cycles. The capacity design problem is modeled as an ILP; all the working paths are provided as inputs for the ILP and thus the optimization is a non-joint optimization problem. We use the following notations, parameters and variables:

$S$  = set of spans ( $s \in S$ ).

$\mathcal{P}$  = set of all cycles eligible for allocation ( $p \in \mathcal{P}$ ).

$R$  = set of unit working paths ( $r \in R$ ). To handle multi-unit requests, it is possible to consider several identical unit paths.

**Input Parameters:**

$c_s$  = cost of span  $k$ .

$$\delta_s^p = \begin{cases} 2 & \text{if span } s \text{ straddles cycle } p, \\ 1 & \text{if span } s \text{ crosses cycle } p, \\ 0 & \text{otherwise.} \end{cases}$$

$\ell_p$  = number of on-cycle spans of  $p$ -cycle  $p$ .

$MU$  = maximum unavailability of any working path after the allocation of  $p$ -cycles.

**Intermediate Variables:**

$$\alpha_s^p = \begin{cases} 1 & \text{if straddling span } s \text{ is protected by } p\text{-cycle } p; \\ 0 & \text{otherwise.} \end{cases}$$

$$\beta_{rs}^p = \begin{cases} 1 & \text{if } p\text{-cycle } p \text{ protects span } s \text{ on path } r; \\ 0 & \text{otherwise.} \end{cases}$$

$|O_r^p|$  = Number of spans in the subset  $O_r^p$ . Similar notation is used for the other sets.

$U_r^p$  = Unavailability of working path  $r$  in  $p$ -cycle (protection domain)  $p$ .

**Output Variables:**

$x_p$  = number of protection units per  $p$ -cycle  $p$ , which is equivalent to the number of copies of the same cycle  $p$ .

$y_s$  = number of spare units placed on span  $s$ .

$U_r$  = Total end-to-end unavailability of working path  $r$ .

### 3.3.1 Minimum Spare Capacity Optimization Model

The objective is to minimize the total spare capacity cost:

$$\min \sum_{s \in S} c_s y_s. \quad (3.10)$$

We have a first set of constraints, one for each span, that guarantees that exactly one  $p$ -cycle will be allocated to protect each working channel on each span, i.e., for every span  $s$  traversed by  $r$ , there is a unique  $p$ -cycle that protects that span against a single failure:

$$\sum_{p \in \mathcal{P}: \delta_s^p > 0} \beta_{rs}^p = 1 \quad s \in r, r \in R \quad (3.11)$$

The number of copies of each cycle which is required for protection against any

single failure ( $x_p$ ), is identified by the maximum number of working channels on a given span which is protected by  $p$  and traversed by at least one working route. Therefore, the number of required copies of each cycle needs to be considered separately for on-cycle and straddling spans as shown in constraints (3.14), where the values of  $x_p^{\text{on-cycle}}$  and  $x_p^{\text{straddle}}$  are given below. For all  $p \in \mathcal{P}$ ,

$$x_p^{\text{on-cycle}} = \max_{k \in S: \delta_s^p = 1} \left\{ \sum_{r \in R: s \in r} \beta_{rk}^p \right\} \quad (3.12)$$

$$x_p^{\text{straddle}} = \left\lceil \frac{1}{2} \max_{s \in S: \delta_s^p = 2} \left\{ \sum_{r \in R: s \in r} \beta_{rs}^p \right\} \right\rceil \quad (3.13)$$

$$x_p = \max\{x_p^{\text{on-cycle}}, x_p^{\text{straddle}}\} \quad (3.14)$$

Combining (3.14) with (3.12) and (3.13), we derive the equivalent constraints (3.15) and (3.16) which can be used in the ILP model:

$$x_p \geq \sum_{r \in R: s \in r} \beta_{rs}^p \quad p \in \mathcal{P}, s \in S : \delta_s^p = 1 \quad (3.15)$$

$$x_p \geq \frac{1}{2} \sum_{r \in R: s \in r} \beta_{rs}^p \quad p \in \mathcal{P}, s \in S : \delta_s^p = 2 \quad (3.16)$$

The total spare capacity on each span is given by (3.17) and this spare capacity is optimized as described in (3.10):

$$y_s = \sum_{p \in \mathcal{P}: \delta_s^p = 1} x_p \quad s \in S. \quad (3.17)$$

(3.18) constrains the unavailability of the service path to a desired upper limit which is our main objective. The upper limit is an input parameter. The program will allocate  $p$ -cycles, such that this constraint is satisfied. If the desired value for unavailability is too low, a solution may not exist.

$$U_r \leq MU \quad r \in R. \quad (3.18)$$

The end-to-end unavailability of  $r$  can be expressed as follows:

$$U_r \cong \sum_{p \in P} U_r^p \quad r \in R, p \in P. \quad (3.19)$$

Accordingly, we can compute  $U_r^p$  as:

$$\begin{aligned} U_r^p = \{ & |O_r^p| \cdot |O_{\bar{r}}^p| + \frac{1}{2}|O_r^p| \cdot |S_{\bar{r}}^p| + |O_r^p| \cdot |S_r^p| + \frac{3}{4}|S_r^p| \cdot |O_{\bar{r}}^p| + \\ & \frac{1}{2}|S_r^p| \cdot (|S_r^p| - 1) + \frac{1}{2}|S_r^p| \cdot |S_{\bar{r}}^p| + \frac{1}{2}|O_r^p| \cdot (|O_r^p| - 1) \} \cdot U^2 \\ & r \in R, p \in P. \quad (3.20) \end{aligned}$$

Using variables  $\beta_{rs}^p$ , we can compute the number of spans in subsets  $O_r^p$ ,  $O_{\bar{r}}^p$  and  $S_r^p$ , as follows. For all  $r \in R, p \in \mathcal{P}$ :

$$|O_r^p| = \sum_{s \in r: \delta_s^p=1} \beta_{rs}^p \quad (3.21)$$

$$|O_{\bar{r}}^p| = \ell_p - |O_r^p| \quad (3.22)$$

$$|S_r^p| = \sum_{s \in r: \delta_s^p=2} \beta_{rs}^p \quad (3.23)$$

The most challenging part of our model is the enumeration requirement of all spans in  $S_r^p$ . For this purpose, we need another intermediate binary variable  $\alpha_s^p$  which indicates whether a straddling span  $s$  is protected by  $p$ -cycle  $p$  or not. In other words, if  $p$  is allocated for protecting  $s$  traversed by any working path  $r$ , then  $\alpha_s^p$  should be equal to one. Therefore the value of this variable can be found by the following identity:

$$\alpha_s^p = \max_{r \in R: s \in r} \beta_{rs}^p \quad p \in \mathcal{P}, s \in S : \delta_s^p = 2. \quad (3.24)$$

Again, this last identity (3.24) can be transformed into a set of linear inequalities as

follows:

$$\beta_{rs}^p \leq \alpha_s^p \quad p \in \mathcal{P}, s \in r, r \in R : \delta_s^p = 2, \quad (3.25)$$

On the other hand, when a straddling span  $s$  is protected by a cycle  $p$ , there should be at least one path  $r$  that traverses  $s$  and is protected by  $p$ . This is ensured by the following set of constraints:

$$\sum_{r \in R: s \in r} \beta_{rs}^p \geq \alpha_s^p \quad p \in \mathcal{P}, s \in S : \delta_s^p = 2. \quad (3.26)$$

Utilizing the variables  $\alpha_s^p$ , we can determine  $|S_r^p|$  as follows:

$$|S_r^p| = \sum_{s \in S: \delta_s^p = 2} \alpha_s^p - |S_r^p| = \alpha_p - |S_r^p| \quad r \in R; p \in \mathcal{P} \quad (3.27)$$

where

$$\alpha_p = \sum_{s \in S: \delta_s^p = 2} \alpha_s^p \quad p \in \mathcal{P} \quad (3.28)$$

$$\alpha_p \in \mathbf{Z}^+ \quad p \in \mathcal{P}. \quad (3.29)$$

After some algebraic manipulations, we obtain:

$$U_r^p = U^2 \left( \ell_p \left( |O_r^p| + \frac{3}{4} |S_r^p| \right) + \left( \frac{\alpha_p}{2} + 1 \right) (|O_r^p| + |S_r^p|) - \frac{1}{2} |O_r^p| \left( |O_r^p| + \frac{|S_r^p|}{2} \right) \right) \quad r \in R, p \in \mathcal{P}. \quad (3.30)$$

### 3.3.2 Linearizing the Quadratic terms

As can be seen, expression (3.30) exhibits some quadratic terms; this is due to the multiplication of expressions (3.21) to (3.27), which yields quadratic expressions to

formulate the path unavailabilities. We next examine each quadratic term in turn.

$$\alpha_p (|O_r^p| + |S_r^p|) = \left( \sum_{s \in S: \delta_s^p = 2} \alpha_s^p \right) \left( \sum_{s \in r: \delta_s^p > 0} \beta_{rs}^p \right) = \sum_{s, s' \in r: \delta_s^p = 2, \delta_{s'}^p > 0} \alpha_s^p \beta_{rs'}^p, \quad (3.31)$$

$$|O_r^p| \times |S_r^p| = \left( \sum_{s \in S: \delta_s^p = 1} \beta_{rs}^p \right) \left( \sum_{s \in r: \delta_s^p = 2} \beta_{rs}^p \right) = \sum_{s, s' \in r: \delta_s^p = 1, \delta_{s'}^p = 2} \beta_{rs}^p \beta_{rs'}^p, \quad (3.32)$$

$$|O_r^p|^2 = \left( \sum_{s \in S: \delta_s^p = 1} \beta_{rs}^p \right)^2 = \sum_{s \in r: \delta_s^p = 1} \beta_{rs}^p + 2 \sum_{s, s' \in r: \delta_s^p = \delta_{s'}^p = 1} \beta_{rs}^p \beta_{rs'}^p. \quad (3.33)$$

In order to linearize (3.31), we introduce the variables:

$$\hat{\gamma}_{rs s'}^p = \alpha_s^p \beta_{rs'}^p \quad s, s' \in r, r \in R : \delta_s^p = 2, \delta_{s'}^p > 0,$$

and the constraints:

$$\hat{\gamma}_{rs s'}^p \leq \alpha_s^p \quad (3.34)$$

$$\hat{\gamma}_{rs s'}^p \leq \beta_{rs'}^p \quad (3.35)$$

$$\alpha_s^p + \beta_{rs'}^p - 1 \leq \hat{\gamma}_{rs s'}^p \quad (3.36)$$

for all  $s, s' \in r, r \in R : \delta_s^p = 2, \delta_{s'}^p > 0$ . To linearize (3.32) and (3.33), we introduce the variables:

$$\tilde{\gamma}_{rs s'}^p = \beta_{rs}^p \beta_{rs'}^p \quad s, s' \in r, r \in R : \delta_s^p, \delta_{s'}^p \in \{1, 2\},$$

and the constraints:

$$\tilde{\gamma}_{rs s'}^p \leq \beta_{rs}^p \quad (3.37)$$

$$\tilde{\gamma}_{rs s'}^p \leq \beta_{rs'}^p \quad (3.38)$$

$$\beta_{rs}^p + \beta_{rs'}^p - 1 \leq \tilde{\gamma}_{rs s'}^p \quad (3.39)$$

for all  $s, s' \in r, r \in R : \delta_s^p, \delta_{s'}^p \in \{1, 2\}$ .

## 3.4 ILP Solution and Scalability Issues

The ApC presents major improvement over the model proposed in [2] for the reasons explained in Section 3.2. Nevertheless, it is less scalable; hence, we propose various techniques to overcome the scalability issues.

### 3.4.1 Selection of the $p$ -cycles

The first scalability issue that is common to ApC and the model of [2] comes from the requirement of considering all possible simple cycles. However, the size of the candidate set increases exponentially with the network size; enumerating all candidate cycles leads to a huge number of ILP variables and slows down the optimization process. Several approaches have therefore been designed for preselecting a promising set of candidate cycles [32], [44], [33]. More recently,  $p$ -cycle network design without offline cycle enumeration [43] and dynamic generation of promising cycles using column generation method [52] have been introduced. Clearly, ApC is not scalable if we consider an off-line explicit enumeration of all  $p$ -cycles; in our experiments, we use an exhaustive enumeration of all  $p$ -cycles only for the small network instances and the  $p$ -cycle generator of [32], which is based on the *A Priori p-cycle Efficiency* (AE) metric, leading to a smaller set  $P \subseteq \mathcal{P}$  of  $p$ -cycles, for larger network instances.

### 3.4.2 ILP Heuristic Techniques

The linear model ApC of the previous section has a large number of variables and constraints following the linearization of the quadratic terms. It can be observed in practice that, although very numerous, the linearization constraints (3.34)-(3.39) are easily satisfied. Consequently, we can use the so-called *lazy constraints* [18] to

overcome their huge number as follows. We first solve a reduced ApC model, i.e., the ApC model where we have omitted all constraints (3.34)-(3.39). Then, we check if the optimal solution of the reduced ApC satisfies the linearization constraints, and we add to the reduced ApC only those constraints which are not satisfied by the current solution.

Dealing with the large number of variables is more difficult. We address it by developing a round robin scheme. We first solve the ApC model with a restricted number of  $p$ -cycles, say set  $P_1$ , but large enough in order to make sure we have a solution. Let us call  $\text{ApC}(P_1)$  the corresponding model, and denote by  $\tilde{P}_1 \subseteq P_1$  the set of  $p$ -cycles in  $P_1$  that are used in the optimal solution of  $\text{ApC}(P_1)$ , i.e., such that  $x_p > 0$  for  $p \in \tilde{P}_1$ . We then solve the model  $\text{ApC}(\tilde{P}_1 \cup P_2)$  where  $P_2$  is a set of additional  $p$ -cycles. We keep adding new cycles, until we have completed a first round, i.e., we exhausted the set  $P \subseteq \mathcal{P}$  of selected (promising) potential  $p$ -cycles. In practice, we go on with a new round as long as an improvement was obtained in the previous round. At the outset, we order the  $p$ -cycles with respect to the AE metric of [32].

### 3.4.3 Incremental Optimality Gap

Last, in order to avoid generating too large search trees in the branch-and-bound algorithm, we use an incremental gap solution. This means that the ILP is first solved with a low precision (high optimality gap) and then we solve it again, with a higher precision after setting the upper bound to the optimal value obtained in the previous solution.

### 3.4.4 Unavailability Overestimation

In spite of the above features, a selected set of  $p$ -cycles, the lazy constraint technique and a round robin ILP heuristic, it remains difficult to solve the ApC model, except

for very small traffic instances. We therefore propose below a reduced version of the ApC model called “ApC-over”, which leads to an unavailability overestimation. We will show in the experiments that it is fairly accurate. The idea is to bound and approximate the size of subset  $S_{\bar{r}}^p$  with the following equation:

$$|S_{\bar{r}}^p| \cong \bar{\alpha}_p - \sum_{s \in r; \delta_s^p=2} \beta_{rs}^p \quad r \in R; p \in P \quad (3.40)$$

where  $\bar{\alpha}_p$  is the total number of spans which straddle the cycle  $p$ . In other words, we approximate the subset  $S_{\bar{r}}^p$  as the set of all straddling spans in  $p$  except the ones that are on  $r$  and protected by  $p$ . Using this approximation leads to overestimation of unavailability contribution in categories **C-2** and **C-6**, because as illustrated in section 3.2.1, there may be some straddling spans which are not protected by  $p$ ; hence, they will not result in unavailability for this domain. In Section 3.5, we will evaluate this approximation and the amount of overestimation which results from this approach. By using (3.40), there will be no need for intermediate binary variables  $\alpha_s^p$  and the corresponding constraints (3.25) and (3.26). Subsequently, linearization variables  $\hat{\gamma}_{rss'}^p$  disappear together with the linearization constraints (3.34)-(3.36). Therefore, the “ApC-over” model will be significantly easier to solve than ApC model; indeed it can be solved for larger instances without the heuristic techniques introduced in section 3.4.2.

## 3.5 Numerical Results

We evaluate our availability-aware design models on different network scenarios and compare them with the model in [2] (which we call ApC-old) in terms of efficiency, complexity and accuracy. We also investigate the tradeoff between capacity investment and MTTR reduction and its effect on service availability in different network scenarios. We assume that demands are routed in advance using Dijkstra’s shortest

path algorithm. Also, we assume that each span has enough spare channels to support the protection capacity required by the optimal solution.  $U$  is supposed to be equal to  $10^{-3}$ . The solutions for the ILP problems are obtained by implementing the model in C++, using the “CPLEX Concert Technology” and solving using the solver CPLEX 11.0.1 [18]. We stop the branch-and-bound process, when the MIP gap (the gap between the relaxed LP lower bound solution and the incumbent ILP solution) is less than 5%.

### 3.5.1 Capacity Efficiency

In order to evaluate the capacity efficiency, we measure the resource redundancy in different network and traffic instances. Table 3.2 illustrates the resource redundancy which is achieved from the solution of different models for 9n17s (9 nodes, 17 spans) network [3] which has a total of 115 candidate cycles. The first column shows the minimum required availability of each unit demand in the network. The second column is the number of symmetric unit demands on each node pair in the network. In order to individually evaluate the effect of over counting and category **C-7**, we consider the three models (ApC, ApC-over, ApC-old) with and without **C-7**. Due to the very large number of variables and constraints, we are only able to solve the ApC model on 9n17s network with two unit demands per each node pair. In addition, we use the round robin technique explained in 3.4.2 where the set of cycles are divided into intervals of 30 cycles for each iteration. In practice, the best solution of the round robin ILP heuristic is usually achieved in the first interval of the first round and we only need to consider the first round.

As can be seen in Table 3.2, for higher values of the required availability, the solution of ApC model with round robin ILP heuristic significantly outperforms the solution in [2] (ApC-old, which results in overestimating the availability). For instance, when the minimum required availability for each path is set to 99.9988% and

there are two symmetric unit demands on each node pair, the ApC model with category **C-7** achieves a solution with 83.05% redundancy, whereas ApC-old [2] yields a solution with a redundancy of 88.13%. Also, we can observe that the performance of ApC-over model is between the models ApC and ApC-old. This means that, although the ApC-over is an approximate model, it is still more accurate than the model in [2]. Comparing the redundancies with and without **C-7**, we observe that the impact of this new category is significant when higher level of availability is required. For relaxed values of required service availability, the solutions of all models converge. The reason is that the upper bound limitation of unavailability constraints in (3.18) are gradually being removed and the solutions of all models tend to a solution for protecting single failures. Since the ApC-over model has a significantly smaller number of constraints and variables, it can be applied on larger network instances with larger set of traffic demands. Table 3.3 shows the performance comparison of ApC-over and ApC-old [2] on COST239 with 11 nodes and 26 spans. For this network, we selected 254 promising cycles among the whole set of candidate cycles using the AE metric. It can be seen that although the ApC-over model is an approximation of ApC, it still outperforms the solution of the model in [2] for all considered availabilities and network scenarios (1%-4% smaller redundancy).

Table 3.2: Comparing the Redundancy in our models and prior work in [2] for different Availabilities on network instance 9n17s

Availability	Number of Demands	ApC		ApC-over		ApC-old [2]	
		without C-7	with C-7	without C-7	with C-7	without C-7	with C-7
99.9992%	2	101.69	111.02	110.17	111.86	113.56	115.25
	3	-	-	112.43	114.12	113.30	115.25
99.9988%	2	75.97	83.05	78.81	86.44	79.66	88.13
	3	-	-	77.97	85.87	78.53	88.13
99.9984%	2	69.49	69.49	69.49	69.49	71.19	72.88
	4	-	-	69.91	70.76	70.34	71.19
99.9976%	2	66.10	66.10	66.10	66.10	66.10	66.10
99.9970%	2	66.10	66.10	66.10	66.10	66.10	66.10

Table 3.3: Comparing the Redundancy on COST239 network

Availability	Number of Demands	ApC-over		ApC-old [2]	
		without C-7	with C-7	without C-7	with C-7
99.9988%	2	69.94	70.90	71.38	72.10
	3	69.92	70.97	71.23	71.54
99.9984%	2	64.22	64.71	66.70	67.18
	3	64.53	64.67	66.72	66.82
99.9980%	2	61.46	61.64	64.17	64.24
	3	61.09	61.14	64.03	64.20
99.9976%	2	60.09	60.23	61.16	61.38
	3	59.97	60.24	61.17	61.25

As can be seen in Tables 3.2 and 3.3, in all three considered models, the redundancy increases for higher values of minimum required availability. One reason is that, for higher availability, the operator has to allocate smaller cycles for protecting the demands and therefore more copies of cycles are required for providing the desired value of availability along with 100% single failure protection. This issue can be observed in Figure 3.4 which presents the average cycle length for different values of service availability in the two network instances with 3 demands on each node pair. The average cycle length is evaluated as:

$$\bar{\ell} = \frac{\sum_{p \in P} x_p \ell_p}{\sum_{p \in P} x_p} \quad (3.41)$$

It can be seen that the average cycle length gradually decreases for higher values of minimum required availability.

### 3.5.2 Design Complexity

Table 3.4 presents a comparison of the complexity of different models for one network instance (9n17s with 2 unit demands on each node pair and a minimum availability of 99.9988% for each demand). The number of binary variables and the number of constraints are given in the first two rows for each ILP model. The third row is the

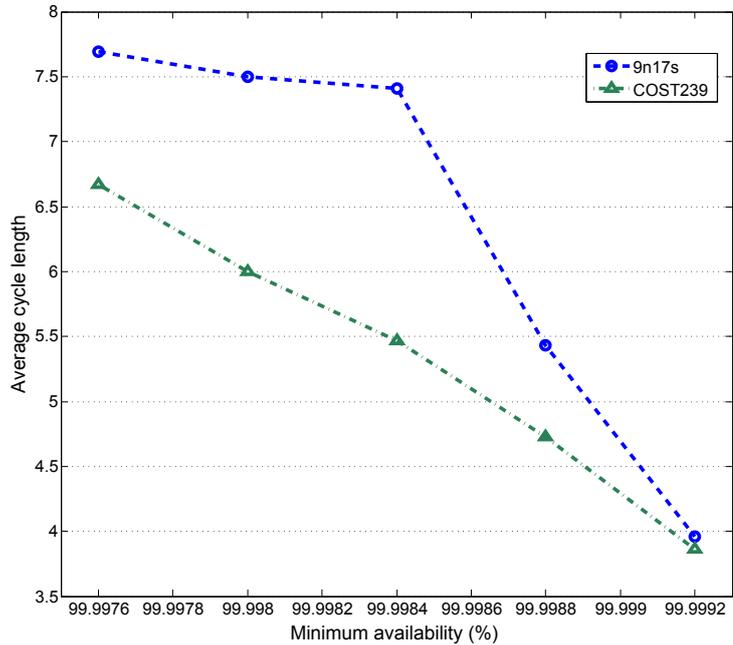


Figure 3.4: Average cycle length vs. minimum required availability for the two network instances with 3 demands on each node pair (ApC-over).

overall number of required Simplex iterations for achieving the optimal ILP solution (with less than 5% accuracy) in the branch-and-bound process. In the last row, we measure the total solution time of the three models. We observe that the ApC model has a substantially large number of constraints and variables; to overcome this difficulty, we applied the round robin method introduced in Section 3.4.2 where the set of cycles is divided into intervals of 30 cycles in each iteration. By using this technique, we can significantly reduce the number of variables and constraints in the solved ILP problems (9,256 binary variables and 23,228 constraints for this instance). However, it should be noted that we need to solve several ILP models each with a subset of 30 cycles until all cycles are considered and no further improvement is achieved. Therefore, the ApC model is prohibitively expensive to be applied on larger network instances. However, in the ApC-over model, solution time and complexity are significantly reduced.

Table 3.4: Comparison of the Complexity of Different Models for One Network Instance

	<b>ApC</b>	<b>ApC-over</b>	<b>ApC-old [2]</b>
#Binary variables	49,741	19,584	8,280
#Constraints	12,9256	37,401	2,188
#Simplex Iterations	252,824,346	281,461	381,707
Solution time (sec)	37,9280	1,255	169

### 3.5.3 Accuracy

One metric for evaluating the accuracy of the solutions is to determine the “precision gap”, which is the gap between the objective value of the achieved ILP solution and the relaxed LP solution of the ApC model. Figure 3.5 illustrates the precision gap of the different models for varying availability in 9n17s network with 2 demands on each node pair. It can be seen that for a low availability (99.9976%), the solution of all models have the same precision. The reason is that for this value of availability, unavailability constraints are not effective. The inaccuracy of round robin heuristic is significant for higher values of availability in the ApC model. In addition, it is clear that although the ApC-over model is an approximation of ApC model, it is still more accurate than ApC-old.

As discussed in Section 3.4.4, in the ApC-over model, all straddling spans, either protected or not, are taken into account, whereas in ApC model only those straddling spans which are protected by the corresponding protection domain are considered. This source of inaccuracy for ApC-over model is illustrated in Table 3.5. Therein, the number of unit copies of the cycles, total number of straddling spans and the number of straddling spans which are protected by the cycle are given for all the cycles which are used in the solution. We observe that for this network instance with the availability of 99.9988% on each demand, in most of the cases the straddling spans are protected by the cycle (for those cycles which have nonzero number of straddling spans). For example, one unit of cycle  $C_0$  is used in the solution and this cycle has 6 straddling spans which are all protected by  $C_0$ , whereas the cycle  $C_{11}$  has two copies

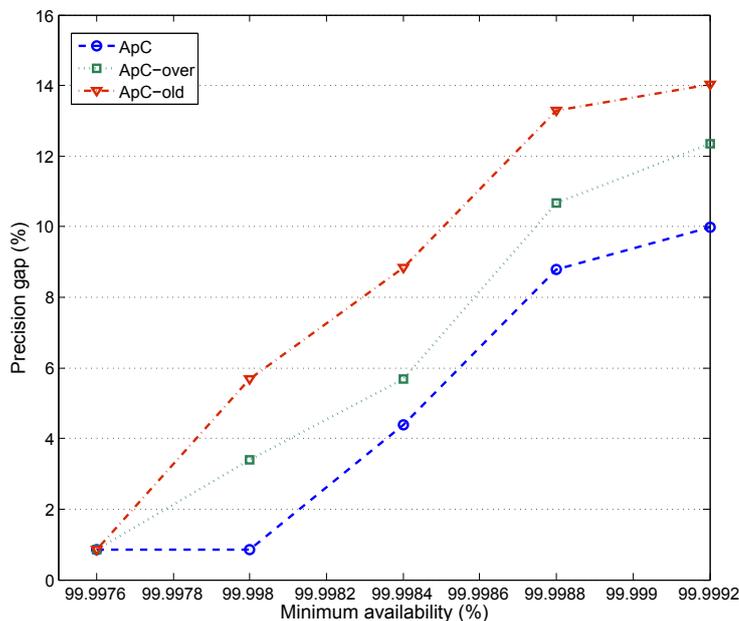


Figure 3.5: Precision gap of three models for different values of required service availability in 9n17s network with 2 demands on each node pair.

in the solution and 6 straddling spans, 5 of which are protected by  $C_{11}$ .

### 3.5.4 Tradeoff between MTTR and Redundancy

The authors of [65] showed by numerical experiments that there is a tradeoff between capacity investment and MTTR (Mean Time To Repair) reduction efforts for achieving high service availability in networks designed to be 100% restorable against single failures. Here, by applying the ApC-over model on different network instances, we evaluate the relation between reducing outage time and resource redundancy for fixed levels of service availability. As reported in [65], with a reasonable approximation, the value of MTTR can be shown to be directly proportional to the physical unavailability of each span ( $U$ ). Up until now, we have fixed the value of  $U$  to  $10^{-3}$ . Now, we investigate the effect of assigning different values to  $U$  (which in essence is equivalent to varying the outage time of the span) on the achieved redundancy for the two test networks. Figure 3.6 presents the amount of redundancy by varying the physical unavailability of each span when the minimum service availability is set to

Table 3.5: The amount of overestimation in ApC-over for 9n17s network with 3 demands on each node pair and minimum availability of 99.9988%

Cycle index	# copies of the cycle ( $x_p$ )	# of straddling spans ( $\bar{\alpha}_p$ )	# of self-protected straddling spans ( $S^p$ )
$C_0$	1	6	6
$C_5$	2	1	0
$C_6$	2	1	0
$C_{11}$	2	6	5
$C_{13}$	1	4	3
$C_{15}$	8	0	0
$C_{16}$	1	4	3
$C_{20}$	2	0	0
$C_{21}$	2	0	0
$C_{24}$	1	0	0
$C_{26}$	1	0	0
$C_{27}$	2	1	1
$C_{28}$	2	1	1
$C_{29}$	1	0	0

99.99% for the two networks (9n17s and COST239), both with 2 demands between each node pair. For other traffic scenarios, the behavior and most of the values would be the same in those network instances. In this figure,  $U$  is varied with a step of  $10^{-3}$ . We observe that for network 9n17s, for  $U$  equal to  $10^{-3}$  and  $2 \times 10^{-3}$ , the same redundancy is obtained to achieve the required end to end service availability. In other words, no additional resources are required to be deployed by the network operator, even if the span outage probability (hence period) doubles. Alternatively, varying the value of  $U$  from  $4 \times 10^{-3}$  to  $3 \times 10^{-3}$  (i.e., if the network operator spends more time and money in reducing the MTTR), a substantial save (close to 47%) in protection resources can be achieved for the same end to end service availability. This hence could be used as an economic guideline for a network operator to decide where and how they should allocate their budget. For higher values of availability, we have to limit the value of  $U$ , since our ApC-over model may not achieve a solution when the unavailability constraints are too tight. Since the physical unavailability of each

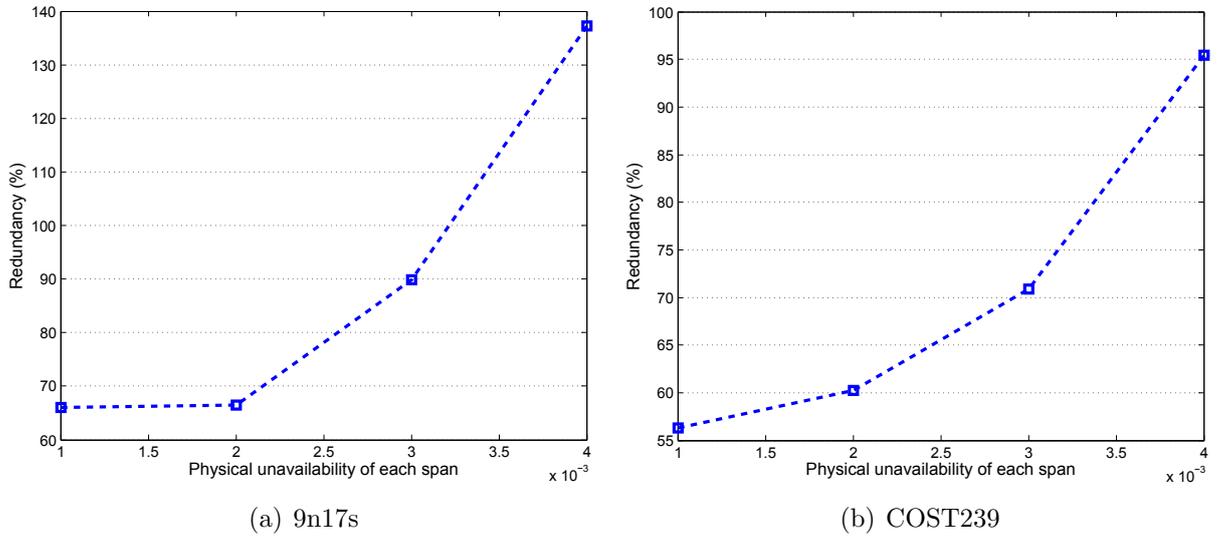


Figure 3.6: Resource redundancy for varying values of physical unavailability of each span; the minimum service availability 99.99%

span ( $U$ ) can also be in the range of  $10^{-4}$  in long-haul networks [3, 100], it would be desirable to investigate this tradeoff for smaller values of  $U$ . Figure 3.7 illustrates the redundancy for varying values of  $U$  in the range of  $10^{-4}$  when the minimum service availability is set to 99.999%. The relation between redundancy and physical unavailability is almost the same as in the previous figure.

### 3.6 Conclusion

We presented enhanced models for availability-aware provisioning in  $p$ -cycle based networks. Our approach builds upon previous work in [2] and we resolved two main flaws in the prior work to achieve an exact non-joint optimization model for the service provisioning problem. One flaw was the over-counting in availability subsets and the other was overlooking a category (C-7) of dual failures which results in service outage. Then we addressed several techniques for speeding up achieving the solution and dealing with the scalability issue. Our results indicate that for higher values of the required availability, the effect of resolving over-counting the

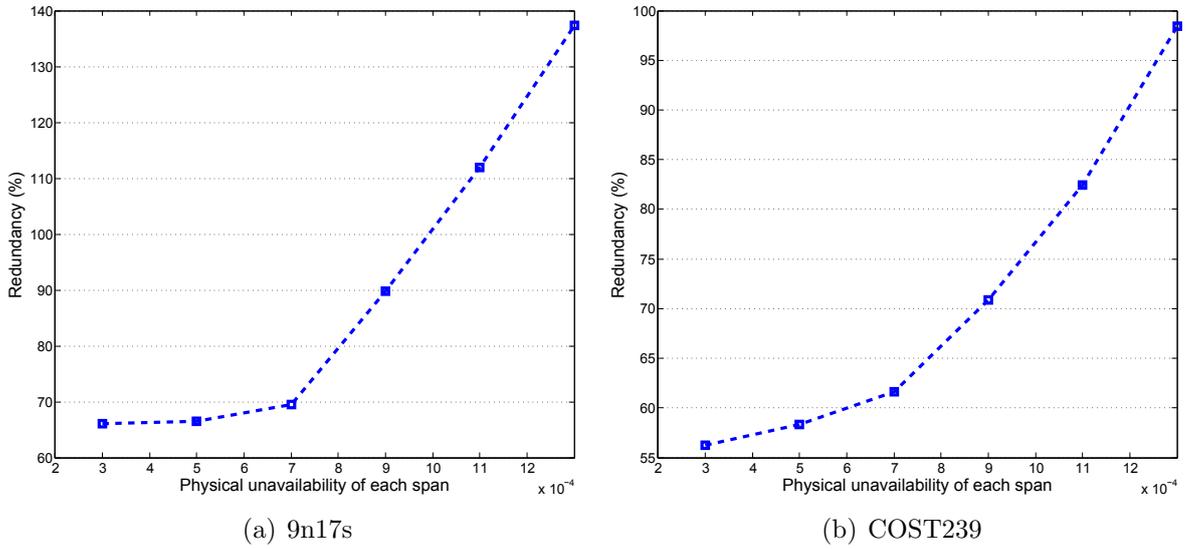


Figure 3.7: Resource redundancy for varying values of physical unavailability of each span; the minimum service availability is set to 99.999%

availability sets is more significant and therefore our model requires less capacity investments from network operators in comparison to prior work. Comparison of the redundancies with and without considering **C-7** shows that the effect of this new category is remarkable when higher level of availability is required. We further analyzed the tradeoffs between reducing the MTTR and deploying more protection capacity to achieve a certain service availability. This study helps network operators wisely allocate their budget to maintain a certain level of service guarantees in their network.

## Chapter 4

# Network Protection Design Models Using Pre-Cross-Connected Trails

In this chapter, we investigate the capability of  $p$ -trails in protecting traffic demands in a mesh-based survivable network. By taking the sharing capability of  $p$ -trails into account, we introduce optimization models to verify the remarkable efficiency of  $p$ -trails. We derive two ILP models for survivable network design using  $p$ -trails. In our first model, the optimal solution is obtained from a candidate set constructed by exhaustive enumeration of all simple trails. We observe that the size of our ILP model, and therefore the computation time, become prohibitively large making the model unpractical for larger network instances. Therefore, to overcome this scalability issue, we develop a better model for this complex optimization problem using a primal-dual decomposition of the original problem based on the column generation (CG) optimization method. Our developed design approach is shown to be very scalable, as opposed to other prior  $p$ -trail design methods; further, we show that  $p$ -trails are more efficient than  $p$ -cycles in terms of resource redundancy in the network.

The rest of this chapter is structured as follows. We present in Section 4.1 the problem statement and motivate the work by some illustrative examples. The ILP model for survivable network design based on  $p$ -trails is given in Section 4.2. We

introduce our CG model for obtaining the overall optimal solution consisting simple and/or non-simple trails and cycles in Section 4.3. The numerical results are given in Section 4.4. Finally, we conclude this chapter in Section 4.5.

## 4.1 Motivation and Problem Statement

As explained in Section 2.3 and illustrated in Figure 2.5, two  $p$ -trails can share their protection capacity on a common sub-trail, if there is no “branch point” on the common links along the protection paths. Accordingly, we present the following definition which identifies the conditions where two  $p$ -trails can share a protection unit on a subtrail.

**Definition 1** Let  $t_1 = (a - b)$  and  $t_2 = (c - d)$  be two distinct  $p$ -trails (the letters indicate the end nodes of the trails). The trails  $t_1$  and  $t_2$  can synchronously share the trail  $(v - w)$  if one of the following conditions is true:

1.  $(v - w)$  is a subtrail of both  $t_1$  and  $t_2$ , such that  $v = c$  and  $w = d$
2.  $(v - w)$  is a subtrail of both  $t_1$  and  $t_2$ , such that  $v = a$  and  $w = b$
3.  $(v - w)$  is a subtrail of both  $t_1$  and  $t_2$ , such that  $v = c$  and  $w = b$
4.  $(v - w)$  is a subtrail of both  $t_1$  and  $t_2$ , such that  $v = a$  and  $w = d$

In summary, we note that the trail  $v - w$  (or equivalently  $w - v$ ) can be shared by  $t_1$  and  $t_2$ , if the end nodes of  $v - w$  are identical to either end nodes of one of the trails.

Using this definition, we show that more general structures can be constructed by merging simple trails that can only protect the links coinciding to their end nodes. In other words, considering the sharing capability and by merging the simple trails, we can construct (originally not enumerated) non-simple trails with sub-cycles. This argument is explained via the illustrative example in Figure 4.1. The set of unit

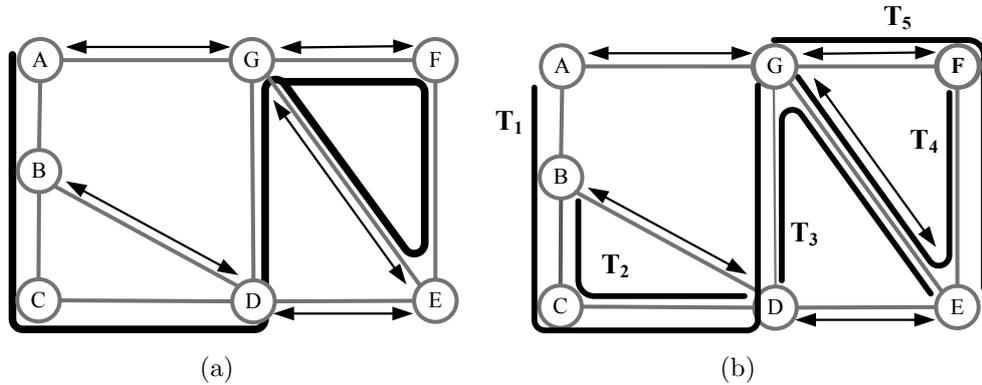


Figure 4.1: An example of a network protected by a set of  $p$ -trails

demands on links  $AG$ ,  $BD$ ,  $DE$ ,  $EG$ , and  $FG$  can be protected by the non-simple trail depicted in Figure 4.1(a). As can be observed in Figure 4.1(b), the non-simple trail in Figure 4.1(a) can be decomposed to its elementary simple trails that can only protect the links with the same end nodes as the trail. That is,  $T_1$  is protecting  $AG$ ,  $T_2$  is protecting  $BD$ ,  $T_3$  is protecting  $DE$ , and so on. We note that according to condition (1) or (2) in Definition (1), trail  $T_2$  shares both of its protection units on links  $BC$  and  $CD$  with trail  $T_1$ . Similarly, trail  $T_3$  shares its protection unit on link  $DG$  with trail  $T_1$  and the protection unit on link  $EG$  with  $T_4$  based on condition (3) or (4). Finally, the protection unit on link  $EF$  is shared between trails  $T_4$  and  $T_5$ . The non-simple trail in Figure 4.1(a) is therefore constructed by merging the elementary trails  $T_1 - T_5$ .

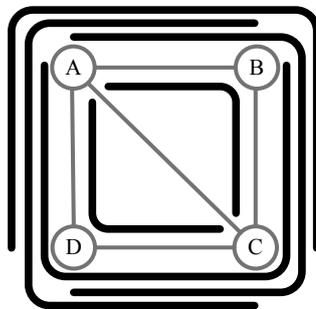


Figure 4.2: A simple  $p$ -cycle constructed by merging the underlying elementary simple trails

Another interesting point is that the protection capacity of every simple cycle

can be explored by merging the underlying elementary simple trails. For instance, merging (i.e., superimposing) the set of six elementary trails depicted in Figure 4.2 yields the construction of the cycle  $(A - B - C - D - A)$  with the same protection capability, i.e., one protection unit for on-cycle links and two protection units for the straddling link  $AC$ .

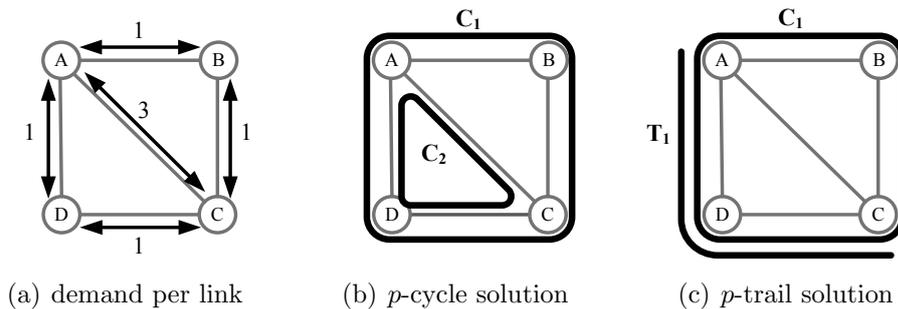


Figure 4.3: Survivable network design based on  $p$ -cycles and  $p$ -trails

Next, we compare the efficiency of designing survivable networks based on  $p$ -trails and  $p$ -cycles. Note that in the optimal solution of  $p$ -cycle based network, the protection capacity of some  $p$ -cycles may not be fully utilized. This occurs mainly when the traffic demands or the costs per link are distributed non-uniformly [101]. In such cases,  $p$ -trail network design may improve the efficiency by removing the cyclic constraint on the protection structures. This feature is illustrated in Figure 4.3 for a very small network instance with the traffic demands as given in Figure 4.3(a), i.e., one demand on links  $AB$ ,  $BC$ ,  $CD$ , and  $AD$  and three demands on link  $AC$ . The optimal solution for  $p$ -cycle design contains two cycles  $C_1$  and  $C_2$  as shown in Figure 4.3(b) with a total of 7 protection units. We observe that the protection unit on link  $AC$  is wasted, since no demand is actually being protected using this link. This can be avoided by choosing trail  $T_1$  instead of  $C_2$  as shown in Figure 4.3(c). This means that the solution of  $p$ -trail design contains cycle  $C_1$  and trail  $T_1$  with a total of 6 protection units.

## 4.2 ILP Formulation

We develop an ILP model for the optimal design of single failure protection of mesh networks using  $p$ -trails. Our objective is to find the minimum cost spare capacity allocation such that every unit demand is protected against single link failures. We assume that the routing of the demands is done in advance using a standard shortest path algorithm. We use the following notations, parameters and variables:

### Sets:

$L$  = set of links in the network.

$\mathcal{T}$  = set of (not necessarily distinct) simple candidate trails eligible for allocation.

### Input Parameters:

$c_\ell$  = protection cost per spare channel on link  $\ell$ .

$w_\ell$  = number of working units on link  $\ell$ .

$$\alpha_\ell^t = \begin{cases} 1 & \text{if the trail } t \text{ can protect a demand on the link } \ell, \\ 0 & \text{otherwise.} \end{cases}$$

$$\gamma_\ell^t = \begin{cases} 1 & \text{if the trail } t \text{ passes through the link } \ell, \\ 0 & \text{otherwise.} \end{cases}$$

$$\beta_\ell^{t_1, t_2} = \begin{cases} 1 & \text{if trails } t_1 \text{ and } t_2 \text{ can not share a protection unit on the common link } \ell, \\ 0 & \text{otherwise.} \end{cases}$$

### Intermediate Variables:

$$y_\ell^{t_1, t_2} = \begin{cases} 1 & \text{if trails } t_1 \text{ and } t_2 \text{ do not share the same protection unit on link } \ell, \\ 0 & \text{otherwise.} \end{cases}$$

### Output Variables:

$$x_t = \begin{cases} 1 & \text{if the trail } t \text{ is selected in the solution,} \\ 0 & \text{otherwise.} \end{cases}$$

$z_\ell$  = number of protection (spare) units placed on link  $\ell$

In this model, the set of candidate trails is obtained by exhaustive enumeration of all possible paths between the end nodes of every link. As explained in Section 4.1, each trail can protect one unit demand on the link that has the same end nodes as the candidate trail. Therefore, in our ILP model, the input parameter  $\alpha_\ell^t$  is equal to one, when the trail  $t$  has the same end node as the link  $\ell$ . Moreover, since we are dealing with unit copies of candidate trails, each trail should be enumerated in the solution space as many times as the number of demands on its end nodes. Hence, the solution space may contain non-distinct trails depending on the traffic distribution.

The parameter  $\beta_\ell^{t_1, t_2}$  is identified based on Definition 1; i.e., if two trails pass one of the conditions in Definition 1 and they are not copies of the same trail then  $\beta_\ell^{t_1, t_2} = 0$ , otherwise, it is set to 1. Note that in our model, the copies of the same trail should not share their protection units; i.e.,  $\beta_\ell^{t_1, t_2} = 1$ , when  $t_1$  and  $t_2$  are copies of the same trail. This is a requirement to avoid underestimating the needed protection capacity. For example, in order to protect 3 unit demands on link  $AC$  in Figure 4.3, the model should choose one copy of the trail  $(A - B - C)$  and two distinct copies of trail  $(A - D - C)$  which are not sharing their protection capacity with each other.

The objective of our ILP model is to minimize the total cost of the spare capacity:

$$\min \sum_{\ell \in L} c_\ell z_\ell. \quad (4.1)$$

Constraint (4.2) ensures that there are enough trails in the solution to protect all demands against any single failure.

$$\sum_{t \in \mathcal{T}} \alpha_\ell^t \cdot x_t \geq w_\ell \quad \ell \in L \quad (4.2)$$

In order to determine the sharing capability among the set of trails, we define the

binary variable  $y_\ell^{t_1, t_2}$  that expresses the conflict of two chosen trails  $t_1$  and  $t_2$  on the common link  $\ell$ . This variable is determined in constraint (4.3), which implies that if two trails  $t_1$  and  $t_2$  are chosen in the solution and according to Definition 1, they can not share a protection unit on the common link  $\ell$ , then the variable  $y_\ell^{t_1, t_2}$  should be set to one.

$$y_\ell^{t_1, t_2} \geq \beta_\ell^{t_1, t_2} (x_{t_1} + x_{t_2} - 1) \quad \ell \in L, t_1, t_2 \in \mathcal{T} \quad (4.3)$$

$$z_\ell \geq \gamma_\ell^{t_1} x_{t_1} + \sum_{t_2 \in \mathcal{T}} y_\ell^{t_1, t_2} \quad \ell \in L, t_1 \in \mathcal{T} \quad (4.4)$$

Constraint (4.4) identifies the number of required protection units on each link. This constraint is for avoiding branch points and states that if trail  $t_1$  passing through link  $\ell$  is chosen in the solution, there should be one extra protection unit for every other chosen trail  $t_2$  which is in conflict with  $t_1$  on link  $\ell$ . In other words, if two chosen trails are branching at one node, there should be distinct protection units for each of them on their common links. For example, if the two trail in Figure 2.5(a) where supposedly chosen in the solution,  $y_\ell^{T_1, T_2}$  would be one on the common link  $CE$ ; hence, there would be two protection units on this link.

Constraints (4.2), (4.3), and (4.4) constitute our first ILP model for spare capacity allocation using  $p$ -trails. In order to improve the scalability of the ILP model, we only consider the variables  $y_\ell^{t_1, t_2}$  and the corresponding constraints when trails  $t_1$  and  $t_2$  traverse through the link  $\ell$  and  $\beta_\ell^{t_1, t_2} = 1$ . This way, we can decrease the number of required computation in the process of obtaining the ILP solution; however, the model has major scalability issues.

In Table 4.1, we evaluate this model on three different network instances. For each instance, we measure the number of candidate trails, nonzero  $\beta$  parameters, variables and the constraints in the model. This table reveals that the ILP model

Table 4.1: Evaluating the ILP model on different network instances

<b>Network</b>	<b>#candidate trails</b>	<b>#nonzero <math>\beta</math></b>	<b>#variables</b>	<b>#constraints</b>
6n11s	275	31,772	32,058	32,811
10n16s	434	205,150	205,600	207,992
9n17s	1,177	1,101,916	1,105,110	1,110,558

has a large number of constraints and variables which prohibits obtaining the ILP result in a reasonable amount of time, whereas the corresponding solution of the  $p$ -cycle model can be obtained in a few seconds for these network instances. For example, in the network instance 10n16s with 10 nodes and 16 links [1], this ILP model obtains a result with an optimality gap<sup>1</sup> of 52.77% in 45,050 seconds. Clearly, the computational cost of this model is too high and indeed a more scalable model is required for designing  $p$ -trail based protected networks. A scalable model would facilitate the design of larger networks (size and traffic instances) more efficiently and would allow network operators to quickly reconfigure their resources and protection patterns, for instance upon network element failure or any change in the traffic being routed in the network. Reconfiguration helps in protecting newly arrived demands and increases the potential to protect subsequent failures after the occurrence of first failure [57].

### 4.3 A Column Generation Design Model

In order to overcome the difficulty of achieving the overall optimal solution, we use the Column Generation (CG) decomposition algorithm. The idea of the CG algorithm is to only generate the variables when needed, i.e., when the reduced cost of a variable is negative [46]. The CG algorithm gives the optimal solution by generating only a fraction of the possible variables which are implicitly enumerated. In this decomposition technique, the linear programming problem is divided into “master” and

---

<sup>1</sup>Also known as MIP gap is the gap between the relaxed LP lower bound solution and the incumbent ILP solution

“pricing” sub-problems. The master problem contains a restricted set of variables with the main objective subject to some of the original constraints. Master problem gives a relaxed linear programming (LP) solution where the integrality constraints of variables are removed. The objective of the pricing problem is to minimize a so-called “reduced cost” and to pass a new promising column to the restricted set in the master problem. This procedure iterates until no negative reduced cost can be obtained in the pricing problem, and therefore the master problem yields the optimal solution with a restricted set of variables [49]. The interested reader is referred to [49], where the optimality of LP relaxed solution of the restricted master model is proven using duality theorem.

In our CG model for  $p$ -trail based network design, the master problem involves the main objective which is to minimize the total spare capacity while providing 100% single failure restorability for all demands on every link. The master problem starts from a feasible solution and passes the dual variables ( $\theta_\ell$ ) to the pricing model. In the pricing subproblem, a new column is generated which corresponds to a trail and its corresponding protection capability. The generated trail can be a non-simple trail or cycle which is constructed by choosing a set of links in the network that yield a negative reduced cost. The new generated columns are added to the master problem, until no further negative reduced cost can be obtained. In the last iteration, the integer solution of the master problem is obtained to identify the required amount of protection capacity on each link.

### 4.3.1 Master Problem

We derive the LP model for the master problem using the following notations:

**Sets:**

$L$  = set of links in the network.

$\mathcal{C}$  = set of candidate trails eligible for allocation including the structures generated

in the pricing subproblem.

**Input Parameters:**

$c_t$  = protection cost of the unit trail  $t$ .

$w_\ell$  = number of working units on link  $\ell$ .

$a_\ell^t$  = number of protection units provided by trail  $t$  on link  $\ell$

**Output Variables:**

$z_t$  = number of copies of the unit trail  $t$

The objective of the master problem is to minimize the total protection cost.

$$\min \sum_{t \in \mathcal{C}} c_t z_t. \quad (4.5)$$

where  $c_t$  is the total protection cost of the unit trail  $t$ , which is the summation of protection cost of the links passing through  $t$ , that is,  $c_t = \sum_{\ell \in L} \gamma_\ell^t c_\ell$ .

The only constraint of the master problem is to guarantee the protection of every working unit against any single link failure.

$$\sum_{t \in \mathcal{C}} a_\ell^t z_t \geq w_\ell \quad \ell \in L \quad (4.6)$$

The input parameter  $a_\ell^t$  identifies the protection relationship between trail  $t$  and link  $\ell$ , i.e., the number of protection units provided by trail  $t$  for link  $\ell$ . As will be seen in Section 4.3.2, the protection capability of the generated structures is identified in the pricing subproblem.

### 4.3.2 Pricing Problem

The objective of the pricing problem is to minimize the reduced cost of the master problem and construct new promising candidate  $p$ -trails to be added to the restricted

set of trails in the master problem. We derive the ILP model for the pricing problem, using the following notations:

**Sets:**

$L$  = set of links in the network.

$\mathcal{T}$  = set of distinct simple trails eligible for allocation.

**Input Parameters:**

$c_\ell$  = protection cost per spare channel on link  $\ell$ .

$$\gamma_\ell^t = \begin{cases} 1 & \text{if the trail } t \text{ passes through the link } \ell, \\ 0 & \text{otherwise.} \end{cases}$$

$$\alpha_\ell^t = \begin{cases} 1 & \text{if the trail } t \text{ can protect a demand on the link } \ell, \\ 0 & \text{otherwise.} \end{cases}$$

$\theta_\ell$  = dual variable for link  $\ell$  associated with constraint (4.6).

**Output Variables:**

$$x_\ell = \begin{cases} 1 & \text{if link } \ell \text{ is spanned by any chosen trail in the solution,} \\ 0 & \text{otherwise.} \end{cases}$$

$$y_t = \begin{cases} 1 & \text{if the trail } t \text{ is selected in the solution,} \\ 0 & \text{otherwise.} \end{cases}$$

$p_\ell$  = number of protection units provided for link  $\ell$

As mentioned earlier, the pricing problem attempts to minimize the reduced cost, which can be written as in (4.7), where  $(\theta_\ell)$  is the dual variable corresponding to constraint (4.6) of the master problem.

$$\min \sum_{\ell \in L} (c_\ell x_\ell - \theta_\ell p_\ell). \tag{4.7}$$

Constraint (4.8) implies that there should be a protection unit on link  $\ell$ , when it traverses through any trail  $t$  which has been selected in the solution.

$$x_\ell \geq \gamma_\ell^t y_t \quad \ell \in L, t \in \mathcal{T} \quad (4.8)$$

$$y_{t_1} + y_{t_2} \leq 1 \quad t_1, t_2 \in \mathcal{T} : \max_{\ell \in L} \{\beta_\ell^{t_1, t_2}\} = 1 \quad (4.9)$$

$$p_\ell \leq \sum_{t \in \mathcal{T}} \alpha_\ell^t y_t \quad \ell \in L \quad (4.10)$$

Constraint (4.9) prevents the pricing model from choosing the conflicting trails. That is, if two trails have a branch point on any of their common links, they should not be selected at the same time. This constraint states that for every two trails  $t_1$  and  $t_2$ , if there exist any link ( $\ell$ ) in the network where  $\beta_\ell^{t_1, t_2} = 1$ , then these two trails can not be chosen at the same time.

Constraint (4.10) says that the number of protected working units on link  $\ell$  can not be more than the number of selected trails in the solution that can potentially protect this link. Note that in the pricing model, we have the same assumptions as before that a trail can only protect the links that have the same end nodes as the trail. Using this assumption, the protection capability of a constructed structure can be obtained from constraint (4.10) according to the discussion in Section 4.1. When the optimal value of the objective or the reduced cost in (4.7) is negative, the pricing model generates a new promising column corresponding to a protection structure  $t_i$  and passes it to the master subproblem by setting the corresponding parameters identifying the traversed links and the protection capacity of the generated trail; i.e.,  $\gamma_\ell^{t_i} = x_\ell$  and  $a_\ell^{t_i} = p_\ell$ .

### 4.3.3 Discussion

As explained in Section 4.1, the protection capacity of every simple cycle can be explored by merging the underlying elementary simple trails. Therefore, the master

and pricing models given in Sections 4.3.1 and 4.3.2 can find a solution consisting of open-ended  $p$ -trails and simple  $p$ -cycles whose protection capacity have been fully explored. However, there might be cases where the protection capacity of a non-simple  $p$ -cycle is not fully explored by merging the underlying simple trails. In fact, this happens when a non-simple  $p$ -cycle figure-eights itself nodewise as illustrated in Figure 4.4. The non-simple  $p$ -cycle in Figure 4.4(a) provides one protection unit for its on-cycle links and two protection units for the straddling links  $AD$  and  $BC$ ; but, this protection capacity can not be exploited by decomposing the cycle into its elementary trails. Figure 4.4(b) shows that the decomposed trails for providing two protection units on the straddling link  $BC$  are in conflict with the trails protecting one unit on links  $AB$  and  $CD$ . The reason is that there will be a branch point on node  $E$  and these trails can not merge to form a protection structure with the same protection capacity as the non-simple cycle in Figure 4.4(a).

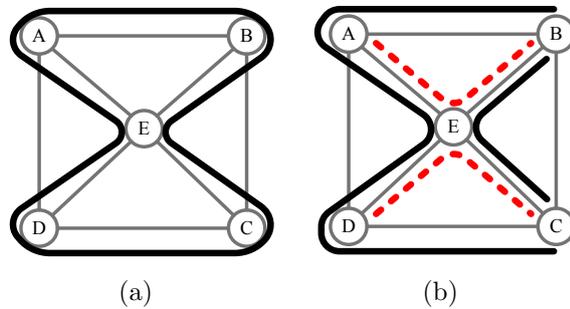


Figure 4.4: A non-simple cycle whose protection capacity cannot be explored by merging the underlying simple trails

To circumvent this problem, we develop yet another pricing model for generating cyclic structures whose protection capacity is fully explored, regardless if they are simple or non-simple cycles. We define the following sets, parameters, and variables for this pricing subproblem:

**Sets:**

$L$  = set of links in the network.

$\mathcal{V}$  = set of nodes in the network.

$\omega(v)$  = set of incident links to the node  $v$ .

$\mathcal{E}(\ell)$  = set of end nodes of the link  $\ell$ .

$\mathcal{J}(V')$  = the set of links such that one of their end-nodes belongs to  $V' \subset \mathcal{V}$  and the other node belongs to  $\mathcal{V} \setminus V'$

**Input Parameters:**

$c_\ell$  = protection cost per spare channel on link  $\ell$ .

$\theta_\ell$  = dual variable for link  $\ell$  associated with constraint (4.6).

**Output Variables:**

$$x_\ell = \begin{cases} 1 & \text{if link } \ell \text{ is spanned by the chosen cycle in the solution,} \\ 0 & \text{otherwise.} \end{cases}$$

$$s_\ell = \begin{cases} 1 & \text{if link } \ell \text{ is straddled by the chosen cycle in the solution,} \\ 0 & \text{otherwise.} \end{cases}$$

$$n_v = \begin{cases} 1 & \text{if node } v \text{ is traversed by the chosen cycle,} \\ 0 & \text{otherwise.} \end{cases}$$

$p_\ell$  = number of protection units provided for link  $\ell$

$u_v$  = an integer variable required for constructing a cyclic path traversing through node  $v$ .

The objective is to minimize the reduced cost the same as the objective of the pricing problem in Section 4.3.2.

$$\min \sum_{\ell \in L} (c_\ell x_\ell - \theta_\ell p_\ell). \quad (4.11)$$

Constraint (4.12) implies that the number of incident links to each node on the

constructed cycle should be a multiple of two.

$$\sum_{\ell \in \omega(v)} x_\ell = 2u_v \quad v \in \mathcal{V} \quad (4.12)$$

Note the variable  $u_v$  can get any integer value; yet, if the generated cycle is simple it can be either zero or one.

Each link can either be a straddling or an on-cycle link with respect to the constructed cycle; that is:

$$s_\ell + x_\ell \leq 1 \quad \ell \in L \quad (4.13)$$

Constraint (4.14) says that a link  $\ell$  can straddle a cycle only if it has at least two on-cycle links adjacent to each of the end nodes of  $\ell$ :

$$2s_\ell \leq \sum_{\ell' \in \omega(v)} x_{\ell'} \quad \ell \in L, v \in \mathcal{E}(\ell) \quad (4.14)$$

If both end nodes of a link are traversed by the constructed cycle, it should be either a straddling or an on-cycle link. This statement can be translated to the following ILP constraint:

$$s_\ell + x_\ell \geq n_{v_1} + n_{v_2} - 1 \quad \ell \in L, v_1, v_2 \in \mathcal{E}(\ell) \quad (4.15)$$

Constraint (4.16) means that a node is chosen, if any of its incident links are spanned by the constructed cycle.

$$n_v \geq x_\ell \quad v \in \mathcal{V}, \ell \in \omega(v) \quad (4.16)$$

The number of protected units on each link (one for on-cycle links and two for straddling links) can be obtained through constraints (4.17) to (4.19) as follows:

$$p_\ell \geq x_\ell \quad \ell \in L \quad (4.17)$$

$$p_\ell \geq 2s_\ell \quad \ell \in L \quad (4.18)$$

$$p_\ell \leq x_\ell + 2s_\ell \quad \ell \in L \quad (4.19)$$

Constraint (4.20) is a variation of sub-tour elimination constraints, stating that if a link straddles a cycle, then its two end-nodes should belong to the same cycle [54].

$$s_\ell \leq \sum_{\ell' \in \mathcal{J}(V')} x_{\ell'} \quad V' \subset \mathcal{V}, 3 \leq |V'| \leq |\mathcal{V} - 3|, \ell \in \mathcal{J}(V') \quad (4.20)$$

Associating this pricing model with the master model in Section 4.3.1, we can construct all the promising cyclic structures whose protection capacity have been fully explored. In the process of finding the optimal solution, we first solve the master problem with the above-mentioned pricing model in order to populate the solution space with cyclic protection structures whose protection capacity have been fully explored. When there is no more negative reduced cost for this pricing model, then we associate the pricing model in Section 4.3.2 to the master model, in order to further construct any generally shaped protection structure. In summary, by considering these two pricing models, we can obtain the overall optimal solution consisting of non-simple trails and cycles.

## 4.4 Numerical Results

We evaluate our CG model for  $p$ -trail network design on various network instances as listed in Table 4.2. This table summarizes the characteristics of each network, including the number of nodes, links, simple cycles and simple trails which are counted

by enumerating all the existing paths between the end nodes of each link excluding the link itself. The first two sample networks are the ones considered in [1] with the same distribution of traffic demands as depicted in Figure 4.5. The rest of the networks are larger and denser samples taken from [3]. For our CG model, we need an initial set  $\mathcal{C}_0$  to solve the master model and start iterating between master and pricing subproblems. To this end, we first extract the cycles chosen in the  $p$ -cycle ILP model for the corresponding network scenario. Then, we construct the initial set by adding these cycles to the set of all simple trails; i.e.,

$$\mathcal{C}_0 = \mathcal{T} \cup \mathcal{P} \quad (4.21)$$

where  $\mathcal{P}$  is the set of cycles chosen in the  $p$ -cycle ILP model.

Table 4.2: Characteristics of Sample Networks

Network	# nodes	# links	# simple cycles	#distinct simple trails
8n14s	8	14	56	300
10n16s	10	16	52	328
9n17s	9	17	131	718
11n20s	11	20	105	607
11n23s	11	23	307	1321

In our first set of experiments, we compare the performance of our CG model with  $p$ -cycle solution and the model presented in [1] in terms of protection cost and running time. The  $p$ -cycle solution is found by implementing the ILP model presented in [36] for  $p$ -cycle network design with 100% single failure restorability. The results of the ILP model in [1] are directly extracted from this reference. Note that for the work of [1], we consider the solutions where each link is spanned at most once, because as stated in that article, traversing a link more than once in the protection path has several drawbacks, such as the need for multiple wavelength convertors and much higher complexity.

For our column generation-based design model, we consider two scenarios. One is

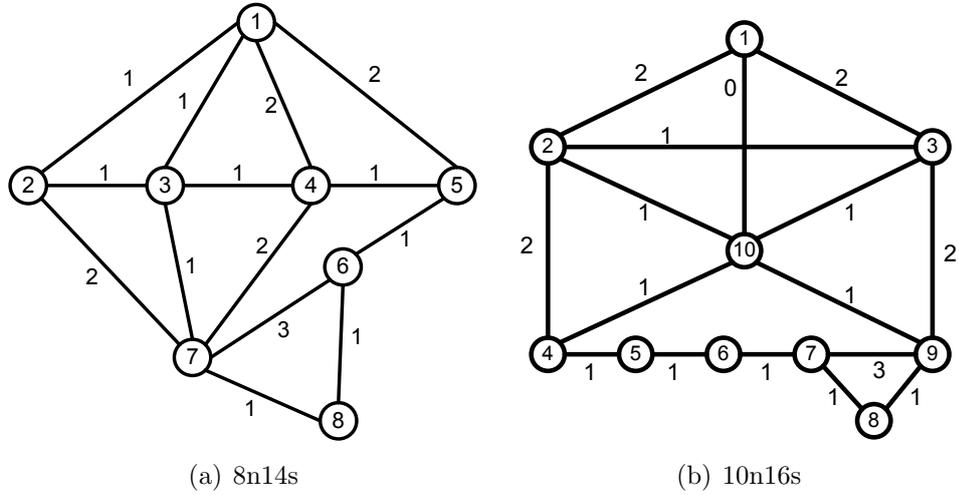


Figure 4.5: Network instances considered in [1]

called  $M\&P_1$  and associates the master model with the first pricing model in Section 4.3.2. The other scenario is to sequentially combine the two pricing models, i.e. to first solve the master model with the second pricing in Section 4.3.3. When there is no more negative reduced cost for this pricing, we associate the first pricing model to the master model in order to find the remaining promising protection structures. We call this scenario as  $M\&(P_1 + P_2)$ .

For the first set of our experiments, we consider the network scenarios in [1] depicted in Figure 4.5, where the traffic demands on each link is shown, and the cost per link is assumed to be unity. The results of applying different protection methods are listed in Table 4.3. The running times of different methods are given in the second column. Note that according to expression (4.21), the running time of our CG models should contain the required time for obtaining the optimal result for the  $p$ -cycle model. We observe that the running time of our CG models is significantly reduced compared to that of the ILP model in [1]. The reduction in running time is more remarkable in  $M\&(P_1 + P_2)$  model. This can be explained by observing the last column of Table 4.3 which indicates the number of generated columns in our CG models. For instance, we see that the  $M\&P_1$  model generates 22 columns for

Table 4.3: Comparison of different protection schemes

Network 8n14s			
Protection method	Running Time	Total cost	# Columns
$p$ -Cycle	0.12 sec	13	.....
$p$ -trail in [1]	258.03 sec	11	.....
CG model $M \& P_1$	23.77 sec	13	22
CG model $M \& (P_1 + P_2)$	0.95 sec	11	3
Network 10n16s			
Protection method	Running Time	Total cost	# Columns
$p$ -Cycle	0.47 sec	18	.....
$p$ -trail in [1]	613.49 sec	15	.....
CG model $M \& P_1$	27.05 sec	16	17
CG model $M \& (P_1 + P_2)$	1.97 sec	15	3

network instance 8n14s, whereas the  $M \& (P_1 + P_2)$  model generates only 3 columns all from the second pricing model which is by far a simpler model in terms of the number of variables and constraints. The same holds for network 10n16s. That is the 3 generated columns of  $M \& (P_1 + P_2)$  model for this network correspond to the second pricing problem, and the first pricing problem runs only once with a non-negative value for the optimal reduced cost.

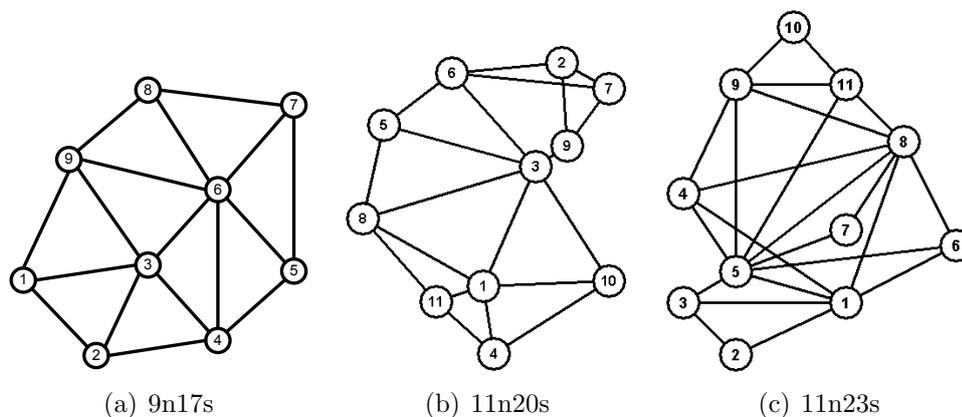


Figure 4.6: Additional selected networks for numerical results

The third column of Table 4.3 indicates the total protection cost of each method.

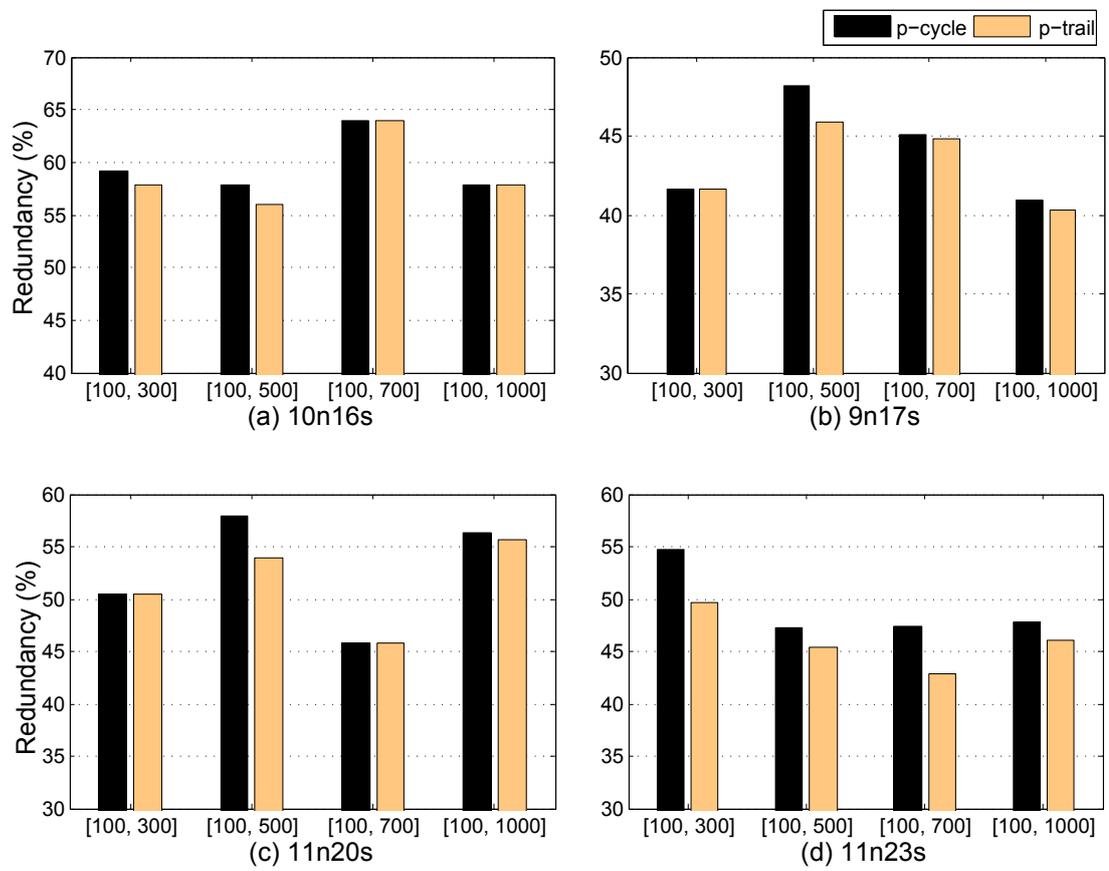


Figure 4.7: Performance evaluation by varying the distribution of cost per link

Table 4.4: Size of ILP Models

<b>Network 8n14s</b>			
<b>Model</b>		<b>#Variables</b>	<b>#Constraints</b>
$p$ -trail in [1]		5,908	7,182
CG model	Master	306	14
	Pricing 1 ( $P_1$ )	328	1,534
	Pricing 2 ( $P_2$ )	58	37,780
<b>Network 10n16s</b>			
<b>Model</b>		<b>#Variables</b>	<b>#Constraints</b>
$p$ -trail in [1]		8,736	10,384
CG model	Master	335	16
	Pricing 1 ( $P_1$ )	360	7,770
	Pricing 2 ( $P_2$ )	68	47,716

For our CG model, the protection cost can be obtained from the objective of the master problem in expression (4.5). We observe that for 8n16s network, the  $M&P_1$  model achieves the same protection cost as the  $p$ -cycle model, which is larger than the protection cost obtained in [1]. This is due to overlooking the non-simple cycles as described in Section 4.3.3. This problem is resolved in  $M&(P_1 + P_2)$  where the optimal solution has the same protection cost as the model in [1]. The same results are obtained for the 10n16s network. In fact the solution of  $M&P_1$  model is a combination of trails and simple cycles, which may or may not be better than the  $p$ -cycle solution. As we observe in both of these network instances, the  $M&(P_1 + P_2)$  model yields the same result as the model in [1] in a much smaller computation time.

We further highlight the advantage of our CG model over the ILP model in [1] by exploring the number of variables and constraints in Table 4.4. It can be observed that the size of the ILP model in [1] is by far larger than the size of master and pricing models in our CG approach. Note that our second pricing model ( $P_2$ ) contains a large amount of constraints; yet, the majority of them are the sub-tour eliminating constraints in 4.20 that are easily satisfied most of the time. Therefore, our CG model significantly reduces the computational cost while yielding the optimal solution. This indeed enables us to evaluate our CG model on larger network instances.

Our second set of experiments consists of evaluating the performance of  $M\&(P_1 + P_2)$  model on different networks by varying the distribution of cost and the number of demands per link. For each network scenario, we compare our CG model with the conventional  $p$ -cycle model in terms of redundancy measured as the ratio of protection cost to the working cost  $\sum_{\ell \in L} c_\ell w_\ell$ . The aim is to observe how the discrepancy of demands and the cost of spare units allow our  $p$ -trail model to outperform the optimal solution of  $p$ -cycle design. Figure 4.7 illustrates the obtained redundancy of our CG model and the  $p$ -cycle model on 4 network instances depicted in Figure 4.5(b) and Figure 4.6. For each network instance, we assume there are three demands on each link. We conduct four experiments on each network, where the cost per link is uniformly randomly distributed in the interval of  $[100, 300]$ ,  $[100, 500]$ ,  $[100, 700]$ , and  $[100, 1000]$ , respectively. It can be seen that the solution of CG model for  $p$ -trail design outperforms that of  $p$ -cycle in most of the cases. In other words, survivable network design based on  $p$ -trails requires less amount of protection capacity compared to  $p$ -cycle-based network design. For example, we recognize in Figure 4.7(d) that for sample network 11n23s, the redundancy decreases from 54.69% in  $p$ -cycle solution to 49.72% in our  $p$ -trail solution when the cost per link is uniformly distributed in the interval of  $[100, 300]$ ; this amounts to 4.97% of improvement in redundancy.

In Figure 4.8, we evaluate the effect of having various traffic demands on each link. We consider the same networks as in Figure 4.7 and carry out four experiments on each network by randomly distributing the number of demands per link in the intervals of  $[1, 3]$ ,  $[1, 5]$ ,  $[1, 7]$ , and  $[1, 10]$ . The cost per link is assumed to be the same on all links. Similar to Figure 4.7, we recognize that having unequal demands per link results in lower redundancy for our  $p$ -trail model compared to the  $p$ -cycle solution in most of the cases. For instance, for sample network 11n23s in Figure 4.8(d), when the number of demands per link is uniformly distributed in the interval of  $[1, 7]$ , our  $p$ -trail solution yields an improvement of 9.89% in resource redundancy

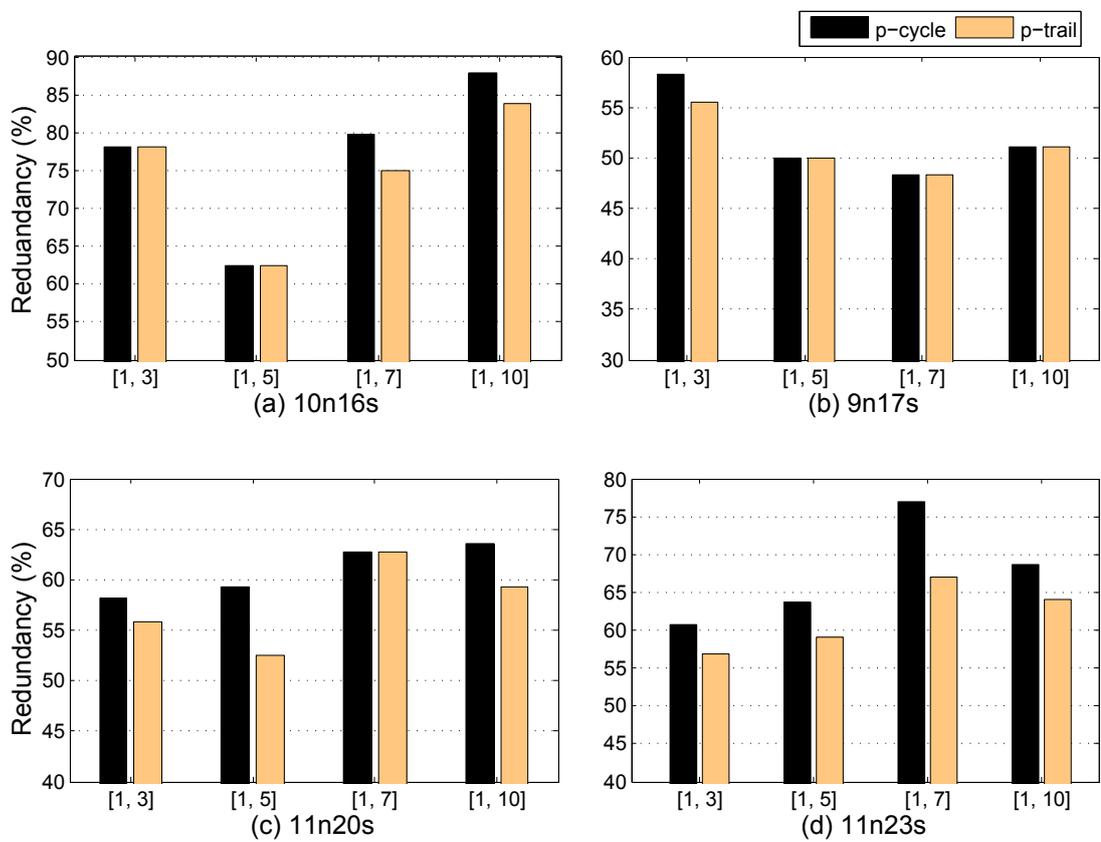


Figure 4.8: Performance evaluation by varying the distribution of the number of demands per link

for this network.

## 4.5 Conclusion

We investigated the survivable network design problem based on pre-cross-connected protection trails, known as  $p$ -trails. We observed that non-simple  $p$ -trails and  $p$ -cycles can be built from merging simple trails. Then, we derived two ILP models for survivable network design with single failure restorability using  $p$ -trails. Our first model is a simple ILP model, where the optimal solution is obtained from a candidate set constructed by exhaustive enumeration of all simple trails. However, as the size of the network increase, the number of candidate trails grows exponentially and therefore the size of our ILP and its computation time become excessive and prohibit us from obtaining solutions for practical networks. To overcome the scalability issue, we derive a second model based on the column generation (CG) decomposition technique, and we show that our CG model is a remarkably scalable ILP model for  $p$ -trail based network design, yielding to an optimal solution with less spare capacity requirement compared to the  $p$ -cycle solution.

# Chapter 5

## Scheduling and Grant Sizing

### Methods for WDM PONs

In most of the previous work, the grant sizing and grant scheduling subproblems have been considered separately, which, as will be shown later, may not achieve optimal network performance. In this chapter, we first revisit the non-joint problem and derive a more efficient ILP model when the bandwidth allocation is pre-determined. Then, we investigate the problem of joint grant sizing and scheduling for multichannel access networks and compare the performance of the joint model with that (non-joint) of [90]. Since the joint model is shown to be hard to solve, except for small network instances, we introduce a Tabu search heuristic for achieving near optimal solutions in a reasonable amount of time.

The rest of this chapter is organized as follows. In Section 5.1, we elaborate on the considered WDM PON network architecture. Section 5.2 presents the problem statement and motivates the work by some illustrative examples. We present the mathematical models for the non-joint and joint optimization problems in Section 5.3 and introduce our Tabu search heuristic in Section 5.4. The numerical results are given in Section 5.5. Finally, we conclude this chapter in Section 5.6.

## 5.1 Network Architecture

We consider a typical PON structure that is comprised of one OLT connecting to multiple ONUs in a tree topology. Two types of widely used ONUs are deployed in our considered architecture; namely the conventional TDM ONUs which operate on a single upstream and a single downstream wavelength and WDM ONUs that can operate on multiple upstream/downstream wavelengths. There are different feasible technologies for realizing WDM ONUs. One promising architecture with evolutionary upgrade path to WDM PONs is depicted in Figure 5.1. In this architecture, the OLT is equipped with an array of ( $W$ ) fixed-tuned receivers and ( $W'$ ) fixed-tuned transmitters for receiving from and sending out data to the ONUs. Two different sets of wavelengths ( $\Lambda_{OLT}^{up}$ ,  $\Lambda_{OLT}^{down}$ ) are used for upstream and downstream transmission. A TDM ONU has only one fixed-tuned transmitter working on wavelength ( $\lambda_k \in \Lambda_{OLT}^{up}$ ) for transmitting upstream data and control traffic to the OLT and one fixed-tuned receiver working on wavelength ( $\lambda_k^* \in \Lambda_{OLT}^{down}$ ) for receiving downstream data and control traffic from the OLT. Conversely, each WDM ONU supports a subset of more than one wavelength (in  $\Lambda_{OLT}^{up}$  and  $\Lambda_{OLT}^{down}$ ) for transmitting and receiving traffic, respectively. Another cost-effective technology for realizing WDM ONUs is to utilize so-called “colorless ONUs” which are wavelength-independent, that is, they are able to use all of the available channels for their upstream and downstream transmissions.

It should be noted that the channel restriction depicted in Figure 5.1 may become an issue regardless of whether colorless or colored ONUs are used. That is, even when an ONU is designed to transmit and receive data on all WDM channels, it may be prevented from operating in this mode. One reason for this restriction is “service separation” where wavelengths are allocated to services based on their quality-of-service requirements, signal characteristics, or tariff structures. Moreover, applying channel restriction to ONUs enables the network operator for flexible leasing of network capacity, in that *the network operator may assign a certain set of wavelength channels,*



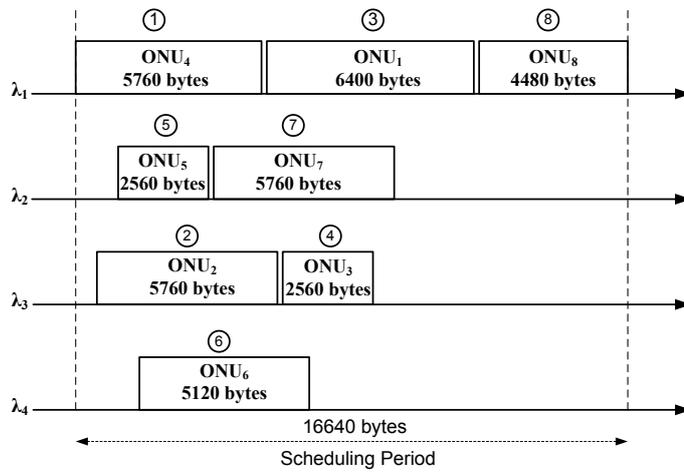
ONU has to wait until the channel becomes available. For example,  $ONU_1$  sends its REPORT request earlier than  $ONU_5$  and therefore  $ONU_1$  is immediately scheduled on channel  $\lambda_1$ ; yet the grant of  $ONU_1$  appears later than that of  $ONU_5$ , because  $ONU_1$  supports only channel  $\lambda_1$  and has to wait until this channel becomes free. The initial gaps on the channels in Figure 5.2(a) represent the time of transmitting the GATE messages of  $ONU_1$ ,  $ONU_5$ ,  $ONU_2$  and  $ONU_6$  after the channels become available.

Table 5.1: Supported channels of ONUs and their corresponding scheduling order

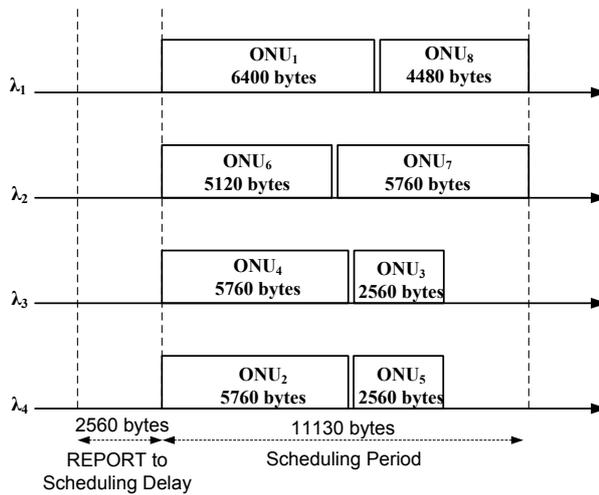
ONU index	Supported channel(s)	Scheduling order	Bandwidth Request (bytes)
1	$\lambda_1$	3	6400
2	$\lambda_3, \lambda_4$	2	5760
3	$\lambda_1, \lambda_3$	4	2560
4	$\lambda_1, \lambda_2, \lambda_3$	1	5760
5	$\lambda_2, \lambda_3, \lambda_4$	5	2560
6	$\lambda_2, \lambda_4$	6	5120
7	$\lambda_2, \lambda_4$	7	5760
8	$\lambda_1$	8	4480

The offline scheduling method is depicted in Figure 5.2(b). The initial gap represents the time which has elapsed between the instant of collecting REPORTs from all ONUs until the scheduling event is started. This time is referred to as the inter-scheduling cycle gap (ISCG) which is determined by the round-trip time (RTT) of the first ONU scheduled on each channel and the computational time for obtaining a schedule [102]. Since we are assuming the same RTT for all ONUs, the ISCG of all channels will be the same. As can be seen in Figure 5.2(b), in offline scheduling where the OLT has received all the ONU REPORT messages and therefore knows the size of requested bandwidth of all ONUs, we can rearrange the ONU grants on the channels to achieve higher channel utilization and lower scheduling cycle length.

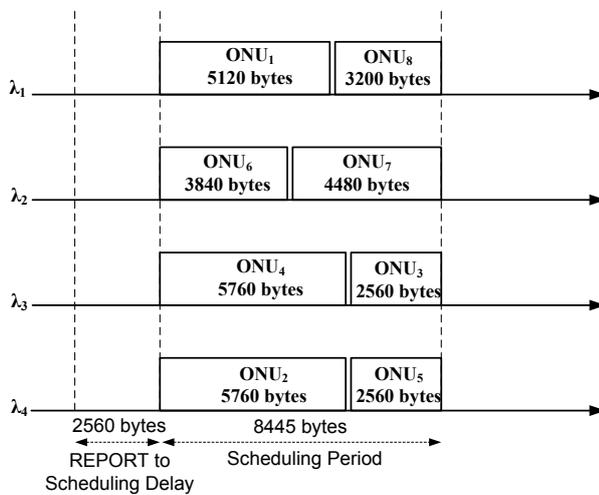
Next, we show that a more efficient framework can be achieved if grant sizing



(a) Online NASC scheduling



(b) Offline scheduling



(c) Joint scheduling and grant sizing

Figure 5.2: An illustrative example for comparing different scheduling methods

and grant scheduling are performed jointly. Assuming that the minimum guaranteed bandwidth for each ONU is 2560 bytes, Figure 2(c) illustrates an efficient joint method with the objective of minimizing the total scheduling period. As one may observe, by reducing the allocated bandwidth for  $ONU_1$ ,  $ONU_6$ ,  $ONU_7$  and  $ONU_8$  by the amount of 1280 bytes, the total scheduling period is remarkably decreased. The allocated bandwidth is reduced and some frames may be delayed at their corresponding queues at the ONUs until the next scheduling cycle. In general, decreasing the scheduling period can increase delay of upstream transmissions. This delay results in the accumulation of buffered data for some ONUs which need to be transmitted during the next scheduling period. Therefore, there is a tradeoff between total scheduling period and scheduling delay for some ONUs. This tradeoff has to be taken into account in the mathematical model by considering a threshold for the maximum allowable delay of each ONU.

## 5.3 Mathematical Formulation

### 5.3.1 Non-Joint Grant Scheduling

The offline grant scheduling problem in evolutionary upgraded multichannel optical access networks is considered as a parallel machine problem with machine (channel) eligibility constraint [90]. According to [93], this problem can be viewed as a special case of unrelated machines in parallel where the processing time (grant size) of a job (ONU)  $j$  is  $P_j$  for the supported machines (channels) and infinity for non-supported machines (channels). Hence, in [90] the authors have formulated the non-joint offline scheduling problem as an ILP model using the following notations:

**Input Parameters:**

$m$  = number of channels

$n$  = number of ONUs

$$p_{ij} = \begin{cases} P_j & \text{if channel } i \text{ is supported by ONU } j; \\ \infty & \text{otherwise.} \end{cases}$$

**Output Variables:**

$$x_{ikj} = \begin{cases} 1 & \text{if position } k \text{ on channel } i \text{ is selected for ONU } j \\ 0 & \text{otherwise.} \end{cases}$$

The objective is to minimize the total completion time:

$$\min \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^n k \times p_{ij} \times x_{ikj} \quad (5.1)$$

subject to

$$\sum_{i=1}^m \sum_{k=1}^n x_{ikj} = 1 \quad \forall j. \quad (5.2)$$

$$\sum_{j=1}^n x_{ikj} \leq 1 \quad \forall i, \forall k. \quad (5.3)$$

Constraint (5.2) ensures that each ONU is assigned to only one scheduling position and constraint (5.3) guarantees that each scheduling position is assigned to no more than one ONU. We call this model “Non-Joint Scheduling with minimizing the Total Completion Time” (NJS-TCT).

We note that in order to achieve better channel utilization and lower queuing delays our objective should be to minimize the length of the polling cycle. However minimizing the total completion time as in (5.1) does not necessarily yield a minimum polling cycle (this statement will be verified by our experiments in Section 5.5). Therefore, we define a new objective for the ILP model as:

$$\min C_{max} \quad (5.4)$$

where  $C_{max}$  is the maximum completion time among all channels or the makespan. For this new objective, constraint (5.5) is added in order to assure that the makespan is greater than the completion time of each channel.

$$C_{max} \geq \sum_{j=1}^n \sum_{k=1}^n k \times p_{ij} \times x_{ikj} \quad \forall i \quad (5.5)$$

The objective (5.4) along with the constraints (5.2), (5.3) and (5.5) form our new ILP model for this problem that we call “Non-Joint Scheduling with minimizing the Makespan” (NJS-M).

### 5.3.2 Joint Scheduling and Grant Sizing

A more general approach for handling this problem is to perform grant sizing and scheduling *jointly*. Unlike the non-joint models in Section 5.3.1 where the size of requested bandwidth of each ONU is assumed to be pre-determined, in our joint model the size of ONU grants are determined along with assigning the wavelength and time slots per ONU requests. The following notations, parameters and variables are used in our joint ILP model:

**Input Parameters:**

$m$  = number of channels

$n$  = number of ONUs

$$\delta_{ij} = \begin{cases} 1 & \text{if channel } i \text{ is supported by ONU } j; \\ 0 & \text{otherwise.} \end{cases}$$

$Q_j$  = the requested bandwidth by ONU  $j$

$B_{min}$  = the minimum guaranteed bandwidth for each ONU

$\eta_j$  = the free buffer size of ONU  $j$

**Intermediate Variables:**

$$y_{ikj} = \begin{cases} P_j & \text{if } x_{ikj} = 1; \\ 0 & \text{otherwise.} \end{cases}$$

$$\alpha_{ij} = \begin{cases} 1 & \text{if channel } i \text{ is allocated for ONU } j; \\ 0 & \text{otherwise.} \end{cases}$$

**Output Variables:**

$C_{max}$  = maximum completion time of all channels or the makespan

$$x_{ikj} = \begin{cases} 1 & \text{if position } k \text{ on channel } i \text{ is selected for ONU } j \\ 0 & \text{otherwise.} \end{cases}$$

$P_j$  = allocated grant size of ONU  $j$

$A_{ij}$  = allocated bandwidth for ONU  $j$  on channel  $i$

Initially, similar to the non-joint model we consider the same objective in (5.4), which is to minimize the makespan. Later, we discuss that more efficient solutions can be achieved by modifying this objective. Constraints (5.2) and (5.3) are the same as for the non-joint problem and (5.5) is re-written as follows:

$$C_{max} \geq \sum_{j=1}^n \sum_{k=1}^n k \times \delta_{ij} \times P_j \times x_{ikj} \quad \forall i \quad (5.6)$$

Note that in our joint model  $P_j$  is a variable that refers to the size of allocated grant to  $ONU_j$ . Thus, constraint (5.6) includes quadratic terms. In order to keep the model linear, we define variables  $y_{ikj} = P_j \times x_{ikj}$  and rewrite (5.6):

$$C_{max} \geq \sum_{j=1}^n \sum_{k=1}^n k \times \delta_{ij} \times y_{ikj} \quad \forall i \quad (5.7)$$

Variables  $y_{ikj}$  are determined in the following set of linearization constraints ( $M$  is a

large positive number):

$$y_{ikj} \leq P_j + M(1 - x_{ikj}) \quad \forall i, k, j; \delta_{ij} = 1 \quad (5.8)$$

$$y_{ikj} \geq P_j - M(1 - x_{ikj}) \quad \forall i, k, j; \delta_{ij} = 1 \quad (5.9)$$

$$y_{ikj} \leq M \times x_{ikj} \quad \forall i, k, j; \delta_{ij} = 1 \quad (5.10)$$

The grant size  $P_j$  of ONU  $j$  can be determined as follows:

$$P_j = \sum_{i=1}^m A_{ij} \quad \forall j \quad (5.11)$$

We assume that the whole request of each ONU is granted on a single channel; i.e., an ONU grant should not be fragmented on multiple channels. In other words, amongst all channels  $i$  only one of the variables  $A_{ij}$  will have a nonzero value which is equal to the allocated bandwidth for ONU  $j$ . To address this issue, we define  $\alpha_{ij}$  which is a binary variable that determines whether channel  $i$  has been allocated for ONU  $j$  or not. The value of  $\alpha_{ij}$  is given by expression (5.12).

$$\alpha_{ij} = \sum_{k=1}^n \delta_{ik} \times x_{ikj} \quad \forall i, j \quad (5.12)$$

Further, while trying to obtain a smaller makespan, the scheduler should ensure a minimum bandwidth for each ONU unless the requested grant size is smaller than the minimum guaranteed bandwidth. To take this matter into consideration, we introduce the following constraints (5.13)-(5.15)

$$A_{ij} = \alpha_{ij} \times Q_j \quad \forall i, \forall j; Q_j \leq B_{min} \quad (5.13)$$

$$A_{ij} \leq \alpha_{ij} \times Q_j \quad \forall i, \forall j; Q_j > B_{min} \quad (5.14)$$

$$A_{ij} \geq \alpha_{ij} \times B_{min} \quad \forall i, \forall j; Q_j > B_{min} \quad (5.15)$$

Formally, the value of the minimum guaranteed bandwidth per ONU ( $B_{min}$ ) is determined by the polling cycle time ( $T_{cycle}$ ); yet, the cycle time is in turn determined by the scheduling algorithm. Therefore, to obtain a reasonable estimate for  $B_{min}$ , we assume  $T_{cycle} = 2ms$  which is a nominal value in PON systems [15]. Then, we estimate  $B_{min}$  as  $(\frac{m}{n}) \times C$  where  $C$  is the typical channel capacity which can be determined as  $\frac{R_{ch} \times T_{cycle}}{8}$  while assuming a typical value for the channel bit rate, e.g.,  $R_{ch} = 1Gb/s$ . It should be noted that by limiting the allocated bandwidth for each ONU in (5.14), some ONUs may experience an increased delay in their upstream transmissions. This will occur when the allocated bandwidth of an ONU is less than the requested bandwidth and the ONU has to truncate a part of the requested grant and postpone it to the next scheduling period. The imposed delay should be limited in order to avoid buffer overflow in each ONU. To this end, we define constraint (5.16) where  $\eta_j$  is the maximum allowable size of accumulated data in ONU  $j$ .

$$Q_j - P_j \leq \eta_j \quad \forall j \quad (5.16)$$

Constraints (5.2), (5.3), (5.7)-(5.15), and (5.16) along with the objective (5.4) constitute our first ILP model for the joint problem that is called ‘‘Joint Scheduling and Bandwidth Allocation’’ (JSBA).

Next, we introduce another model for the joint problem with a new objective. We note that considering the same objective as the non-joint model may increase the idle gap and consequently lead to higher queuing delay and lower channel utilization. This may happen while reducing the size of ONU grants which are being transmitted on the channels whose completion time is less than the makespan. For example in Figure 2(c), the completion time on channels  $\lambda_1$  and  $\lambda_2$  may not be further reduced, because all the scheduled ONUs on these channels have reduced the size of their grants to their buffer limit which is 1280 bytes. Therefore, decreasing the completion time on channels  $\lambda_3$  and  $\lambda_4$ , will not lead to a smaller makespan. In this example, the

grant sizes of  $ONU_2$  and  $ONU_4$  can still be decreased. However, reducing the grant sizes of these ONUs will yield no benefit; it will unnecessarily increase the queuing delay on these ONUs and increase the idle gap at the tail of channels  $\lambda_3$  and  $\lambda_4$  and consequently decrease the average channel utilization. In order to overcome this problem, we modify our objective to jointly minimize the total channel waste as well as the makespan. To this end, we define the channel bandwidth waste  $w_i$  for each channel  $i$  as:

$$w_i = ISCG_i + C_{max} - \sum_{j=1}^n \sum_{k=1}^n k \times \delta_{ij} \times y_{ikj} \quad \forall i \quad (5.17)$$

Assuming the same bit-rate on all upstream and downstream channels, the inter-scheduling cycle gap on channel  $i$  can be expressed as follows<sup>1</sup>:

$$ISCG_i = n \cdot L_{GATE} + \frac{R_{ch}}{8} \sum_{j=1}^n \delta_{ij} \times x_{ij} \times RTT_j \quad \forall i \quad (5.18)$$

where  $L_{GATE}$  is the length of the GATE message transmitted from the OLT to each ONU, and  $RTT_j$  is the round trip time of  $ONU_j$ . The first term in (5.18) is always a constant and the second term refers to the round trip delay of the first ONU which is scheduled on channel  $i$ . This term becomes a constant if we assume that all ONUs have the same round trip delay. As a result, the ISCG will be the same for all channels and can be omitted as a constant from our definition for channel bandwidth waste. Considering the channel bandwidth waste, our new objective can be expressed as:

$$\min \left( C_{max} + \sum_{i=1}^m w_i \right) \quad (5.19)$$

Replacing (5.17) into (5.19) and assuming a constant value for ISCG in all channels,

---

<sup>1</sup>We assume that the computed schedule is repeated on every cycle until the status of the network is changed and hence another schedule is re-computed. Therefore, the computation time of the schedule can be omitted from ISCG.

we can rewrite the objective as:

$$\min \left[ (m + 1) \times C_{max} - \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^n k \times \delta_{ij} \times y_{ikj} \right] \quad (5.20)$$

The objective (5.20) along with the same set of constraints as in the JSBA model comprise our second ILP model, which we call “Modified Joint Scheduling and Bandwidth Allocation” (MJSBA).

## 5.4 Solving The Joint Problem Using Tabu Search

The ILP model developed in Section 5.3.2 is very hard to solve except for small sized network instances (as shown in the next section). The number of variables and constraints in this model become prohibitively large and therefore the computational complexity gets quite high for larger network instances. Thus, it is vital to develop a heuristic in order to get near-optimal solutions in a reasonable amount of time. To this end, we develop a Tabu search method for solving the joint scheduling and bandwidth allocation problem.

Our Tabu search heuristic starts from an initial solution which can be obtained from one of the dispatching rules. Several dispatching rules have been examined in [95] and it was observed that the “largest processing time (LPT) first” dispatching rule yields a reasonable solution for initializing the Tabu search algorithm. One crucial component of the Tabu search algorithm is the choice of the neighborhood. We consider two types of move for our Tabu heuristic. One is reordering and moving the ONU grants from one wavelength to another supported wavelength. In this move, the neighborhood of the current solution is obtained by moving a transmission window of an ONU from its assigned wavelength to another supported wavelength (if there is any). This is similar to insert and pairwise exchange (swap) moves as explained in [103]. In our illustrative example in Figure 5.2(b), one neighbor solution

can be achieved by using this move through exchanging the positions of  $ONU_5$  and  $ONU_7$  on channels  $\lambda_2$  and  $\lambda_4$ . The other move is reducing the transmission window sizes of different ONUs. This move offers more options for swapping ONU grants between different supported channels.

Using these two moves, our Tabu search algorithm performs a local search to explore new feasible solutions. In each iteration of the procedure, both moves are assessed, and the one that yields the best result is chosen as the final move to be performed. Our Tabu method also makes use of a short term memory (Tabu list) that stores information associated with recently explored solutions in order to avoid cycling. For example, the Tabu list contains the positions of the swapped grants, and any move that schedules an ONU grant back to its old position is considered Tabu (i.e., forbidden). Further, and similar to [104], an aspiration criterion which allows to overwrite the Tabu status of a move is used, so that any move that yields better improvement is considered regardless of the status of the move. Search diversification is obtained by allowing the algorithm to make restart and random perturbations. The algorithm restarts after executing  $\lambda$  iterations without any improvement on the current best solution. Periodic random perturbations are also used to enhance the diversification of the search. A perturbation is executed every  $\gamma$  iterations and consists of randomly selecting and executing a move from the neighborhood regardless of its quality and status.

The Tabu search algorithm needs some stopping criteria. One stopping criterion is to iterate for a certain number of iterations depending on the number of ONUs. Furthermore, we note that the percentage of utilization of different channels vary as the grants of ONUs are resized and reordered among different channels. Therefore, we consider another stopping criterion such that the algorithm runs until the last scheduled ONUs on all wavelength channels have the same finishing time. This is equivalent to maximizing the average bandwidth utilization (measured as the ratio

of the sum of ONU transmission times to the total scheduling length of all channels).

## 5.5 Numerical Results

We implemented the ILP models for the joint and non-joint scheduling and bandwidth allocation problem in C++, using the ‘‘CPLEX Concert Technology’’ and solved them using the solver CPLEX 11.0.1 [18]. We used C++ for the implementation of the Tabu search procedure described in Section 5.4. For the number of iterations and perturbation period we used  $\lambda = 200N$  and  $\gamma = 200N$  where  $N$  is the number of ONUs. We consider 4 different network instances  $N_1$ - $N_4$ , by varying the number of ONUs and wavelengths as well as the number of supported wavelengths per ONU. We assume 4 channels and 8 ONUs for network instance  $N_1$ , 4 channels and 16 ONUs for  $N_2$ , 8 channels and 32 ONUs for  $N_3$ , and 10 channels and 64 ONUs for  $N_4$ . In each network instance, we assume that each ONU randomly supports 1, 2 or 3 different wavelengths from the existing set of upstream channels. Other network parameters are shown in Table 5.2.

Table 5.2: Network Parameters

Data rate of WDM wavelengths ( $R_{ch}$ )	1 Gbps
Round trip delay between each ONU and OLT ( $RTT_j$ )	100 $\mu$ s (10km)
ONU buffer size	1 Mbytes
Guard bandwidth between adjacent slots	125 bytes (1 $\mu$ s)
Length of GATE message ( $L_{GATE}$ )	64 bytes

According to the discussion in Section 5.3.2 and assuming a channel data rate  $R_{ch} = 1$  Gb/s (as stated in Table 5.2), we calculate the value of the minimum guaranteed bandwidth ( $B_{min}$ ) for  $N_1, N_2, N_3$  and  $N_4$  as 125000, 62500, 62500, and 39063 bytes, respectively. For the joint scheduling methods, namely the joint Tabu, JSBA, and MJSBA, we assume that the available buffer size of each ONU for reducing the transmission grants ( $\eta_j$ ) is 0.1 Mbytes (10% of the ONU buffer size). Almost in all of our experiments, we observe that assigning a larger value to  $\eta_j$  will not have any

effect on the achieved solution. The reason is that  $\eta_j = 0.1$  Mbytes is large enough to satisfy constraint (5.16) even if the allocated bandwidth of all ONUs is restricted to the minimum guaranteed bandwidth.

### 5.5.1 Different Network Instances

In the first set of our experiments, we evaluate the performance of each scheduling method on different network instances  $N_1$ - $N_4$ . In all cases, we assume that each ONU has a bandwidth requirement randomly uniformly distributed over the interval of  $[0.5B_{min}, 2.5B_{min}]$ . In Table 5.3, the makespans achieved from applying different scheduling methods are shown for our 4 network instances. Table 5.4 presents the corresponding average channel utilization. For computing the average channel utilization, we consider the bandwidth waste of each channel derived from equation (5.17) and the 125 byte guard bandwidth between adjacent slots. As the number of ONUs and wavelength channels increases, the computational complexity of the joint ILP models JSBA and MJSBA rises prohibitively. In particular, the solver is not able to obtain solutions of these two model for network instances  $N_3$  and  $N_4$  even after several days; yet, the results can be achieved for other instances in few seconds.

Table 5.3: Makespan (*msec*) when each ONU has a traffic load in the interval of  $[0.25B_{min}, 2.5B_{min}]$

<b>Network</b>	<b>NJS-TCT</b>	<b>NJS-M</b>	<b>Tabu</b>	<b>JSBA</b>	<b>MJSBA</b>
$N_1$	2.9872	2.6320	2.0926	1.9990	2.0818
$N_2$	3.2469	3.0738	2.4923	1.8149	1.9159
$N_3$	3.5705	3.1592	2.7220	-	-
$N_4$	3.3927	3.1503	2.9077	-	-

The first observation from Tables 5.3 and 5.4 is that our new non-joint ILP model (NJS-M) consistently outperforms the model in [90] (NJS-TCT) for all network instances. For example, in network  $N_3$ , the NJS-M model yields a makespan of 3.1592 msec and average channel utilization of 94%, while these figures stand on 3.5705 msec and 83.17% respectively for the NJS-TCT model (i.e., 11.5% reduction in makespan

Table 5.4: Average channel utilization (%) when each ONU has a traffic load in the interval of  $[0.25B_{min}, 2.5B_{min}]$

<b>Network</b>	<b>NJS-TCT</b>	<b>NJS-M</b>	<b>Tabu</b>	<b>JSBA</b>	<b>MJSBA</b>
$N_1$	79.09	89.35	82.67	82.64	86.50
$N_2$	87.21	91.94	95.18	89.32	90.02
$N_3$	83.17	94.00	94.76	-	-
$N_4$	85.88	92.21	93.52	-	-

and 10.83% increase in average utilization). Second, we notice that the joint methods significantly decrease the makespan in all cases. However, this improvement has been achieved at the cost of reducing the size of allocated bandwidth of some ONUs. For instance, in network  $N_1$  the makespan is reduced from 2.9872 msec in NJS-TCT to 2.0926 msec in Tabu and 1.999 msec in JSBA, but there is an average reduction of 43,855 and 37,142 bytes in the size of allocated ONU bandwidth in the solutions of Tabu and JSBA, respectively.

We also observe that MJSBA slightly increases the makespan; yet it results in higher system utilization when compared to JSBA. For instance, in network  $N_1$ , the solution of MJSBA has a 4% larger makespan than that of JSBA (1.999 msec in JSBA and 2.0818 msec in MJSBA), but the average utilization is 3.86% higher in MJSBA. Furthermore, we can see that for network  $N_2$  the Tabu yields higher utilizations than the joint ILP models. This happens because the Tabu search stops once it achieves a solution where the last scheduled ONUs on all wavelength channels have the same finishing time. Hence, the Tabu will yield a solution with the minimum channel waste which is only incurred by the ISCG gap and guard bandwidth between adjacent slots. We also recognize that our non-joint NJS-M yields higher channel utilization than the joint models. The reason is that by decreasing the makespan in the joint models, the initial ISCG will be more influential. This drawback can be mitigated by reducing the length of ISCG gap, e.g., by omitting the transmission of GATE messages (since the same schedule is repeated, it is not necessary to arbitrate the ONUs on every scheduling cycle).

Table 5.5 presents the computation time of different algorithms when they are performed on the same hardware platform. We see that our Tabu is a promising method for the joint scheduling and grant sizing problem; it provides close to optimal solutions, while significantly reducing the CPU time when compared to the sequential and joint ILP models.

Table 5.5: CPU time (in sec) for different scheduling algorithms

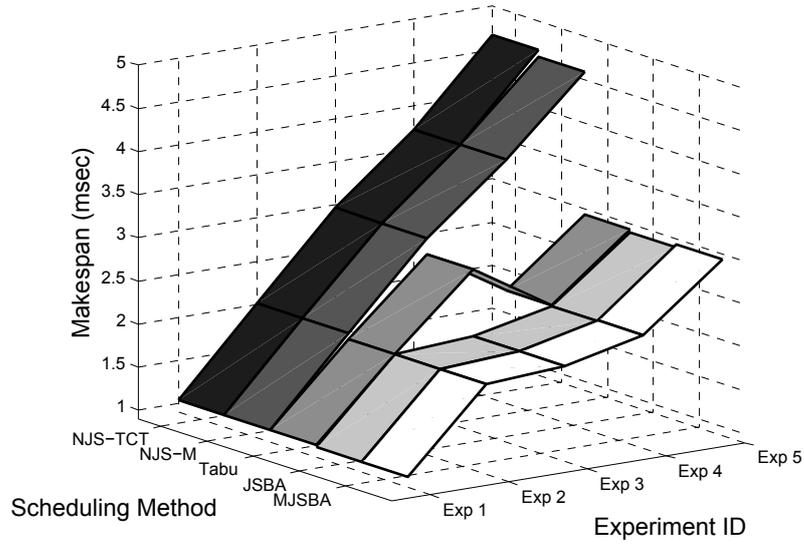
<b>Network</b>	<b>NJS-TCT</b>	<b>NJS-M</b>	<b>Tabu</b>	<b>JSBA</b>	<b>MJSBA</b>
$N_1$	0.0140	0.0230	0.0020	0.2030	0.7729
$N_2$	0.0250	0.0420	0.0030	8.0208	8.6307
$N_3$	0.1060	0.5579	0.0070	-	-
$N_4$	0.7230	6.2880	0.3900	-	-

## 5.5.2 Different Traffic Loads

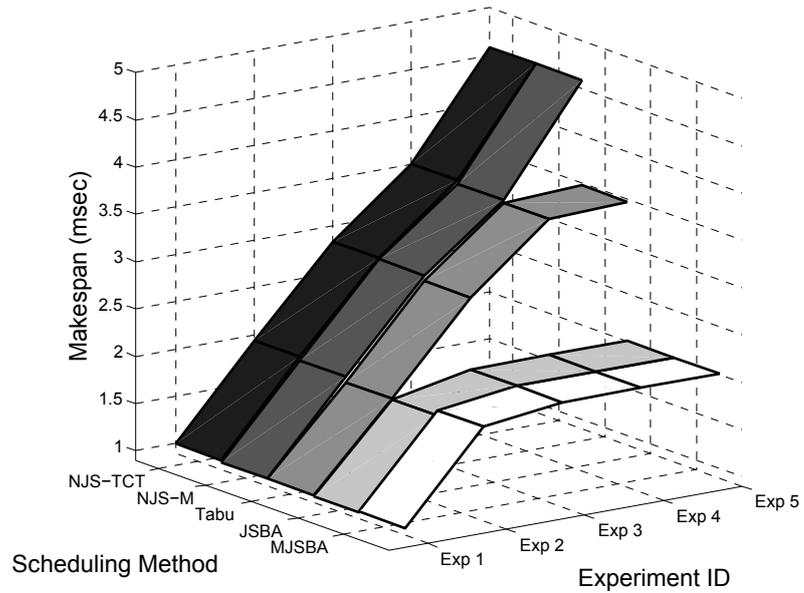
In order to evaluate the effect of different traffic demands on the performance of scheduling methods, we carry out a set of experiments from low to high traffic loads for network instances  $N_1$  and  $N_2$ . Namely, we define the experiments  $E_1$  to  $E_5$  where each ONU has a bandwidth request randomly uniformly distributed over the interval of  $[0.5B_{min}, 0.9B_{min}]$ ,  $[0.9B_{min}, 1.3B_{min}]$ ,  $[1.3B_{min}, 1.7B_{min}]$ ,  $[1.7B_{min}, 2.1B_{min}]$ , and  $[2.1B_{min}, 2.5B_{min}]$  respectively. As can be observed, our experiment configurations cover scenarios from low ( $E_1$ ) to high traffic load ( $E_5$ ).

Figures 5.3 and 5.4 respectively illustrate the makespan and average utilization for different experiments in network instances  $N_1$  and  $N_2$ . As expected, our NJS-M model consistently outperforms the old NJS-TCT model. In Figure 5.3(a), we observe that the Tabu method results in a larger makespan for experiment  $E_3$  than  $E_4$ . This can be explained from Figure 5.4(a) which shows that the Tabu method has the highest channel utilization of 96.12% amongst all methods for experiment  $E_3$  in network instance  $N_1$ . Similar to the results in Table 5.4, in Figures 5.4(a) and 5.4(b), we see that for some instances the non-joint NJS-M achieves higher utilizations than

the joint methods. One interesting observation from Figure 5.3(b) is that for  $E_3$ ,  $E_4$  and  $E_5$  the makespan is equal to the typical value of 2 msec in both JSBA and MJSBA methods. The reason is that both JSBA and MJSBA reduce the transmission bandwidth request of all ONUs to the minimum guaranteed bandwidth, which is derived based on a 2 msec cycle length.

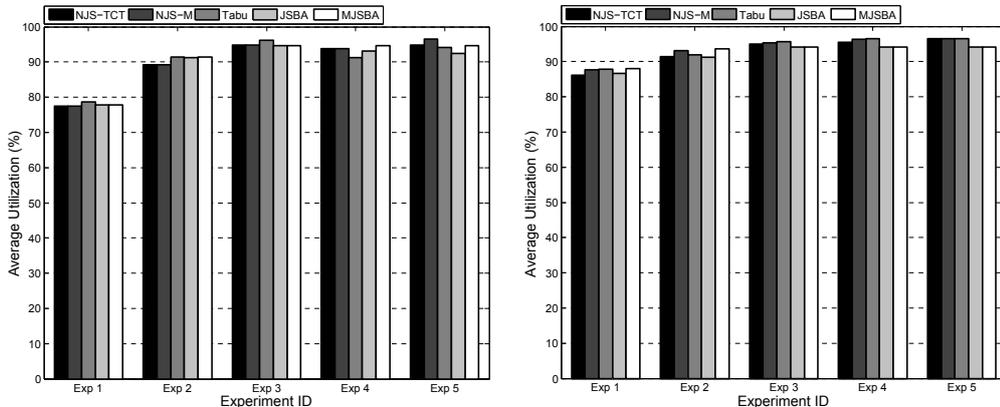


(a) Network instance  $N_1$  with 8 ONUs and 4 wavelengths



(b) Network instance  $N_2$  with 16 ONUs and 4 wavelengths

Figure 5.3: Makespan for different experiment groups



(a) Network instance  $N_1$  with 8 ONUs and 4 wavelengths (b) Network instance  $N_2$  with 16 ONUs and 4 wavelengths

Figure 5.4: Average channel utilization for different experiment groups

### 5.5.3 Packet Level Simulation

We carry out packet-level simulation to study the performance of the proposed scheduling methods; we simulated the operation of the scheduling methods using OMNet++, a discrete event simulator [19]. We assume that the ONU traffic loads are generated at a constant bit-rate (CBR) based on their instantaneous data rates, and the packet size is uniformly distributed between 64 and 1518 bytes. Since the OLT knows the downstream traffic bandwidth demand of each ONU instantaneously, our simulation focuses on bandwidth allocation in the upstream direction. The numerical results are collected for the same set of five experiments  $E_1$  to  $E_5$  in network instances  $N_1$  and  $N_2$ . For each experiment, we take the scheduling solutions from NJS-M, joint Tabu, JSBA, and MJSBA as inputs to our network simulator. The performance metrics are the average packet loss and queuing delay, which are presented in Tables 5.6 and 5.7 respectively.

We observe that for light traffic loads (in  $E_1$  and  $E_2$ ), the joint methods have slightly better performance. In particular, for experiment  $E_2$  in network  $N_2$ , the MJSBA methods yields a 1.98 msec reduction in average queuing delay and a 2.74% reduction in average packet loss rate when compared to NJS-M model. For  $E_2$  in

network  $N_1$ , the packet loss rate is decreased from 8.82% in NJS-TCT to 3.95% in MJSBA (i.e., 4.87% reduction in packet loss rate). In addition, similar to the previous results, our new NJS-M model consistently outperforms the model in [90] for all experiments. We also see that the tradeoff between minimizing makespan and maximizing channel utilization in MJSBA yields a better performance when compared to other joint methods, i.e., Tabu and JSBA. As can be seen in Table 5.7, for experiments  $E_4$  and  $E_5$  the average packet loss is unacceptable for all methods in both network instances. This can be explained by noting the fact that  $E_4$  and  $E_5$  are overloading the network by very high bandwidth requests of ONUs compared to the minimum guaranteed bandwidth. Such overloaded instances should be avoided in practical cases. We also observed (Tables 5.6, 5.7) that when the traffic load is light, online NASC scheduling method provides better performance than our offline algorithms (both in terms of delay and packet loss). The reason is that in online NASC, the OLT does not have to wait for all REPORTs to arrive before making any scheduling decisions. Therefore, on average, each ONU will be granted within a transmission window sooner compared to the offline scheme. However, as the load increases, and for the same reasons stated in Figure 5.2(a), we observe that NASC performance degrades as opposed to our joint scheduling method.

## 5.6 Conclusion

We studied the problem of grant scheduling and bandwidth allocation in evolutionary upgraded WDM PONs. We presented three new ILP models for the non-joint and joint scheduling and grant sizing problem. Since the joint ILP models are very hard to solve, except for small network instances, we introduced Tabu search heuristic for achieving near optimal solutions. Our experiments show that the joint scheduling and sizing algorithms outperform the non-joint models in terms of scheduling cycle length. Deriving a new model for the non-joint problem, we obtained results that

Table 5.6: Average queuing delay (*msec*)

Network $N_1$ (8 ONUs, 4 wavelengths)						
Experiment	NJS-TCT	NJS-M	Tabu	JSBA	MJSBA	NASC
$E_1$	0.95	0.82	0.82	0.81	0.81	0.75
$E_2$	19.36	18.52	18.04	18.13	18.01	16.96
$E_3$	18.53	17.61	17.71	17.35	17.35	17.77
$E_4$	19.67	17.94	18.50	18.13	17.73	18.80
$E_5$	18.45	17.68	18.08	17.99	17.97	19.11
Network $N_2$ (16 ONUs, 4 wavelengths)						
Experiment	NJS-TCT	NJS-M	Tabu	JSBA	MJSBA	NASC
$E_1$	0.75	0.74	0.72	0.71	0.71	0.64
$E_2$	37.77	36.84	36.69	37.03	35.86	33.63
$E_3$	36.11	35.58	35.06	34.75	34.75	34.21
$E_4$	35.00	34.63	34.50	35.72	35.72	35.93
$E_5$	34.90	34.70	34.72	35.86	35.86	36.11

outperform the previous non-joint model in terms of makespan, utilization, queuing delay and packet loss. We further introduced a modified joint model that yields solutions with up to nearly 15% reduction in average queuing delay and 5% reduction in average packet loss. We also conclude that our Tabu search heuristic is a promising solution for the joint scheduling and grant sizing problem. While significantly reducing the computation time compared to the sequential and joint ILP models, our Tabu heuristic provides close to optimal solutions.

Table 5.7: Average packet loss rate (%)

<b>Network <math>N_1</math> (8 ONUs, 4 wavelengths)</b>					
<b>Experiment</b>	<b>NJS-TCT</b>	<b>NJS-M</b>	<b>Tabu</b>	<b>JSBA</b>	<b>MJSBA</b>
$E_1$	0	0	0	0	0
$E_2$	8.82	6.40	4.11	4.53	3.95
$E_3$	38.16	34.92	34.85	33.93	33.93
$E_4$	52.24	47.75	48.95	47.91	46.80
$E_5$	59.23	57.56	58.42	58.14	58.09
<b>Network <math>N_2</math> (16 ONUs, 4 wavelengths)</b>					
<b>Experiment</b>	<b>NJS-TCT</b>	<b>NJS-M</b>	<b>Tabu</b>	<b>JSBA</b>	<b>MJSBA</b>
$E_1$	0	0	0	0	0
$E_2$	6.86	3.52	3.19	4.48	2.53
$E_3$	33.92	32.45	31.99	31.39	31.39
$E_4$	44.72	45.06	43.97	44.18	44.18
$E_5$	57.71	57.48	57.30	58.09	58.09

# Chapter 6

## Scheduling and Bandwidth

### Allocation of 10G-EPON

### co-Existing with WDM-PON

In this chapter, we investigate the problem of optimal scheduling and bandwidth allocation in next generation 10G-EPON coexisting with 1G WDM-PONs. We first propose a network architecture for supporting the coexistence. Then, we derive an ILP model for offline joint scheduling and bandwidth assignment for 10G-TDM and 1G-WDM ONUs. Our goal is to develop efficient bandwidth allocation and scheduling algorithms in this system with multi-rate ONUs. Based on the choice of wavelength channels, the OLT may use separate or the same DBA modules for 1G- and 10G-PONs. To address this fact, we study two scheduling scenarios where the 10G TDM channel is either shared between 1G- and 10G-ONUs, or it is dedicated to 10G-ONUs. We exploit the tradeoff which exists in terms of delay, scheduling length, and channel utilization, when separate or the same DBA modules are used for 1G- and 10G-ONUs. To address the scalability of the ILP model, we introduce a Tabu Search based heuristic for obtaining near optimal solutions in remarkably shorter

computation time.

The rest of this chapter is structured as follows. The network architecture for the coexistence of 1G-WDM and 10G-TDM PON is given in Section 6.1. We present in Section 6.2 the problem statement and motivation of the work by some illustrative examples. We discuss about the bandwidth allocation, delay analysis and present the ILP models for joint scheduling and bandwidth allocation in Section 6.3. The Tabu heuristic is explained in Section 6.4, followed by the numerical results in Section 6.5. Finally, we conclude the chapter in Section 6.6.

## 6.1 Network Architecture

We propose a network architecture for the coexistence of 10G-TDM and future 1G-WDM PONs. As illustrated in Figure 6.1, our PON structure comprises one OLT connecting in a tree topology to multiple 1G- and 10G-ONUs. In order to enjoy the benefits of multi-channel upgraded PON or hybrid WDM-TDM PONs, the upstream transmission waveband should be split into multiple wavelength channels. We note that the upstream waveband for 10G-EPON is too narrow to be split into multiple wavelength, whereas the 100 nm waveband of 1G-EPON can be more easily split into multiple channels for ONU upstream transmissions. Therefore, we consider a dual-rate EPON architecture with 10G-TDM ONUs coexisting with future 1G-WDM ONUs.

One of the most cost-effective technologies for realizing WDM ONUs is to utilize so-called “colorless ONUs” which are wavelength-independent, and make use of a reflective semiconductor optical amplifier (RSOA) at the ONU for remote modulation of the upstream data [85]. In this approach, the OLT is equipped with laser diodes to send optical continuous wave (CW) signals to the attached reflective ONUs, where the CW signal is modulated and sent back to the OLT; hence, no light source is required at the ONU.

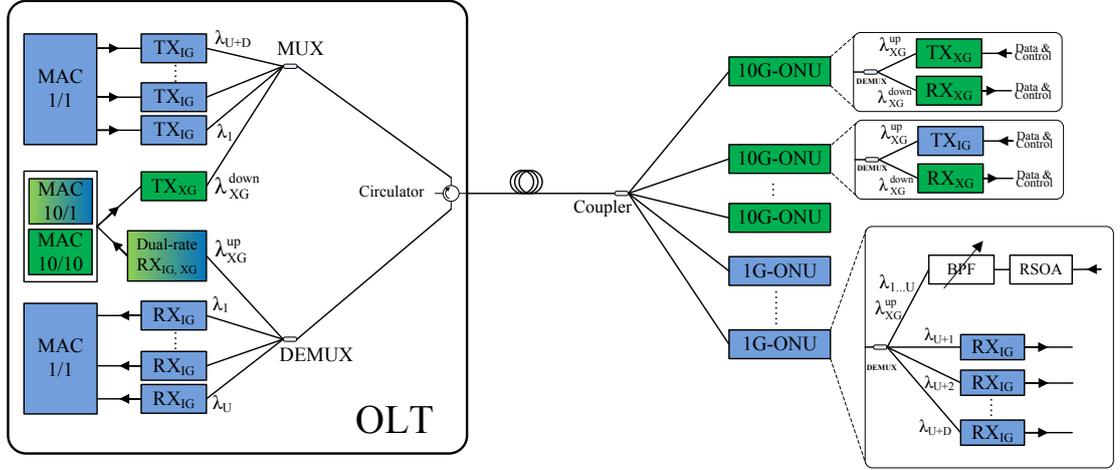


Figure 6.1: Coexistence of 10G-TDM and 1G-WDM PONs

In our architecture, the OLT is equipped with an array of fixed-tuned receivers and fixed-tuned transmitters for receiving from and sending out data to the ONUs. Two types of receivers are deployed at the OLT. One is denoted by  $RX_{IG}$  which is used at one of the upstream channels  $\lambda_1, \dots, \lambda_U$  for receiving data from 1G-ONUs. The other is the dual-rate receiver  $RX_{IG, XG}$  tuned to the center of 10G-EPON upstream waveband ( $\lambda_{XG}^{up} = 1270$  nm) for receiving data from 10G-ONUs and from those 1G-ONUs sharing the upstream channel with 10G-ONUs. Each of the transmitters at the OLT are either fixed tuned to one of wavelengths  $\lambda_1, \dots, \lambda_U$  for sending CW signals to the reflective 1G-ONUs, or they are tuned to one of the wavelengths  $\lambda_{U+1}, \dots, \lambda_{U+D}$  for sending downstream data and control traffic to 1G-ONUs. Also, there is a 10 Gb/s transmitter at the OLT fixed tuned to  $\lambda_{XG}^{down}$  for transmitting downstream data to 10G-ONUs.

The OLT provides three kinds of MAC instances; namely, 1/1 Gb/s, 10/1 Gb/s, and 10/10 Gb/s. The 10G-ONUs are TDM ONUs working on  $\lambda_{XG}^{up}$  and  $\lambda_{XG}^{down}$  channels for their upstream and downstream transmissions, respectively. As shown in Figure 6.1, a given 10G-ONU generates either a 1 or 10 Gb/s signal, depending on which one of the two specified transmit paths is implemented at the ONU [78]. Conversely, the 1G-WDM ONUs are equipped with an RSOA, which can be tuned to all the existing

upstream channels including  $\lambda_{XG}^{up}$  for transmitting upstream data and control traffic to the OLT. This way, the 1G-ONUs are capable of transmitting on all available channels including the 10G channel; we do not however allow simultaneous transmissions on multiple channels. Also, each 1G-ONU employs an array of fixed-tuned receivers, each tuned to one of the wavelengths  $\lambda_{U+1}, \dots, \lambda_{U+D}$  for receiving downstream data and control traffic from the OLT.

## 6.2 Motivation and Problem Statement

It is important to note that even though the upstream waveband for 1G-EPON standard spans 1260 nm to 1360 nm, some network operators may restrict the waveband of 1G-EPON costumers not to extend below 1300 nm in order to avoid inventory problems [77]. Thus, the upstream coexistence can be achieved using WDM. Whether the upstream waveband is restricted or not, 10G-ONUs may or may not share their upstream channel with 1G-WDM ONUs. When the 1G-ONUs and 10G-ONUs operate on the same channel, all ONUs should be controlled by a single scheduler and DBA module at the OLT. Conversely, if the allocated wavebands for 1G and 10G-ONUs are different, the OLT can deploy separate DBA and scheduling modules for 1G and 10G-ONUs. In the following example, we illustrate these two scenarios and their effects on the channel utilization and the length of the scheduling period.

We note that for a lightly loaded network, the online scheduling method of next available supported channel (NASC) provides better scheduling solutions in terms of delay and packet loss compared to offline scheduling methods [90]. On the other hand, it would not be reasonable to share the 10G channel with 1G ONUs when the 10G ONUs are highly loaded. Therefore, we consider a traffic scenario where the 10G-ONUs are lightly loaded and 1G-ONUs have different level of traffic loads from light to heavy. We consider a network with 10 1G-WDM ONUs indexed by  $ONU_{1,2,\dots,10}$  and 2 10G-TDM ONUs indexed by  $ONU_{XG1}$  and  $ONU_{XG2}$ . There are

four 1G WDM wavelengths  $\lambda_1, \dots, \lambda_4$  and one 10G wavelength  $\lambda_{10G}$  for 10G ONUs. The round-trip time (RTT) between each ONU and OLT is assumed to be 100  $\mu\text{sec}$ , which corresponds to a 10 km distance.

The allocated time slots of each ONU are illustrated in Figure 6.2. In Figure 6.2(a), the 10G wavelength channel is dedicated to 10G-ONUs, which are arbitrated according to the online NASC scheduling method. In this case, the OLT polls the 10G-ONUs every 100  $\mu\text{sec}$  and grants the requested bandwidth. The 1G-WDM ONUs are scheduled using the non-joint offline scheduling method presented in [105]. The initial gap represents the inter-scheduling cycle gap (ISCG) which is mainly determined by the RTT of the first ONU scheduled on each channel [102]. We observe that almost 60% of the 10G channel is wasted, whereas this channel could have been utilized more efficiently if it had been shared by 1G-ONUs. In Figure 6.2(b), we see that a more efficient schedule with smaller polling cycle length (or makespan) and higher channel utilization can be obtained when the 10G channel is shared with the 1G-ONUs. As the polling cycle increases for 10G-ONUs, they will have larger bandwidth requests compared to the online scheduling in Figure 6.2(a). Consequently, the average packet delay will increase for 10G-ONUs. To mitigate this problem, we can further reduce the makespan using joint scheduling and bandwidth allocation for 1G-ONUs (Figure 6.2(c)). The transmission window size of highly loaded 1G-ONUs is reduced based on the minimum guaranteed bandwidth of each channel. We observe that in the joint method, the scheduling length and therefore the packet delay for 10G-ONUs are decreased at the expense of a larger delay for the 1G-ONU transmissions. In summary, Figure 6.2 illustrates a clear tradeoff between transmission delay, channel utilization and scheduling period when using different scheduling methods.

## 6.3 Scheduling and Bandwidth Allocation for 10G-TDM and 1G-WDM ONUs

Our goal is to develop efficient bandwidth allocation and scheduling algorithms for the bandwidth requests of 10G-TDM ONUs and 1G-WDM ONUs. We assume that each 1G-ONU can not transmit on more than one channel per cycle. This way, during each polling cycle, the OLT has to send only one CW signal to the WDM-ONU in order to remotely modulate the upstream data; therefore, the planning cost decreases compared to a scenario where the ONU transmissions per cycle are allowed to be bifurcated into different channels and the OLT has to send multiple CW signals to WDM-ONUs in each polling cycle.

### 6.3.1 Bandwidth Allocation

In order to determine the allocated bandwidth for each ONU, we should first determine the minimum guaranteed bandwidth on each wavelength. Namely, we should determine the minimum guaranteed bandwidth for 10G ONUs on 10G channel denoted by  $B_{XG}(\lambda_{XG})$  and that for 1G-ONUs on 1G and 10G channels denoted respectively by  $B_{IG}(\lambda_{XG})$  and  $B_{IG}(\lambda_{IG})$ . Formally, the value of the minimum guaranteed bandwidth is determined by the polling cycle length ( $T_c$ ), which is in turn determined by the scheduling algorithm. Therefore, to obtain a reasonable estimate for the minimum guaranteed bandwidth, we have to assume a typical value for the cycle time, e.g.,  $T_c = 2$  msec. Considering the transmission windows on the 10G channel, we find:

$$T_c = \frac{N_{XG} \cdot B_{XG}(\lambda_{XG})}{R_{XG}} + \frac{n_{IG} \cdot B_{IG}(\lambda_{XG})}{R_{IG}} \quad (6.1)$$

where  $N_{XG}$  is the total number of 10G-TDM ONUs,  $R_{IG}$  ( $R_{XG}$ ) is the effective data rate of 1G (10G) channels, and  $n_{IG}$  is the number of 1G-ONUs which are decided to be scheduled on the 10G channel. We assume that a 10G-ONU can transmit up to

10 times more bytes during a  $T_c$  period than a 1G-ONU, that is:

$$B_{XG}(\lambda_{XG}) \approx 10B_{IG}(\lambda_{XG}) \quad (6.2)$$

Then, we obtain:

$$B_{XG}(\lambda_{XG}) = \frac{T_c}{\frac{N_{XG}}{R_{XG}} + \frac{n_{IG}}{10R_{IG}}} \quad (6.3)$$

The minimum guaranteed bandwidth for the rest of 1G-ONUs on the 1G wavelength channels can be determined as follows:

$$B_{IG}(\lambda_{IG}) = \frac{M}{N_{IG} - n_{IG}} \times (T_c \cdot R_{IG}) \quad (6.4)$$

where  $M$  is the total number of 1G channels, and  $N_{IG}$  is the total number of 1G-WDM ONUs. Next, we determine the allocated bandwidth for each ONU. Let  $Q_j$  be the requested bandwidth and  $P_j$  be the allocated bandwidth for  $ONU_j$ . If the requested bandwidth is less than the minimum guaranteed bandwidth on the assigned channel, then the whole request will be granted, i.e.,

$$P_j = Q_j \quad \forall j : Q_j \leq B_{min}(i, j) \quad (6.5)$$

where  $B_{min}(i, j)$  is the minimum guaranteed bandwidth of  $ONU_j$  on channel  $i$  that can be determined from one of the expressions (6.2), (6.3), or (6.4) (it is obvious that  $B_{min}(i, j) = 0$  for 10G-ONUs on 1G channels).

Otherwise, if the requested bandwidth is greater than the minimum guaranteed bandwidth, the grant size of ONUs will be reduced to meet the following inequality:

$$B_{min}(i, j) \leq P_j \leq Q_j \quad \forall j : Q_j > B_{min}(i, j) \quad (6.6)$$

### 6.3.2 Delay Analysis

In order to obtain a reasonable estimate on the expected transmission delay per each ONU, one needs to know the behavior of the bandwidth requests. To this end, we assume that each ONU request is generated according to its instantaneous data rate  $R_j^u$  on the upstream channel; that is, during each polling cycle, the ONU generates a bandwidth request of  $Q_j = T_c R_j^u$ .

The delay analysis is illustrated in Figure 6.3. The upper axis illustrates the buffer occupancy of ONU  $j$ , while the lower axis shows the data transmission on the upstream channel assigned for ONU  $j$ . In the first cycle, the ONU transmits the packets stored in its buffer and sends its request for the next cycle inside a REPORT message. In the second cycle, the ONU transmits what has been requested (and allocated) in the first cycle; meanwhile, the ONU buffer is being filled with newly generated packets, which will be scheduled for transmission in the third scheduling cycle. Therefore, the data generated at time  $s_j + t_j$  will be transmitted at  $s_j + 2T_j$ , where  $s_j$  is the start time and  $t_j$  is the length of the transmission window allocated for ONU  $j$  on the supported wavelength channel. If the ONU data request is small enough to be transmitted in one cycle, then this would translate into the maximum delay, i.e.,

$$D_j = 2T_c - t_j \quad \forall j : P_j = Q_j \quad (6.7)$$

*Theorem* If the allocated bandwidth is less than the ONU request, the maximum packet delay can be obtained from expression (6.8) in which  $N_j^C$  is the number of polling cycles elapsed until the buffer of ONU  $j$  is full.

$$D_j = (N_j^C + 2) T_c - t_j - N_j^C \left( \frac{P_j}{R_j^u} \right) \quad \forall j : P_j < Q_j \quad (6.8)$$

**Proof** If the granted bandwidth is less than the request, some packets will stay in the ONU's buffer and will be transmitted in subsequent cycles. As shown in Figure

6.3, the data which is generated at time  $s_j + t_j + P_j/R_j^u$  has to be transmitted at  $s_j + 3T_c$ . This leads to a delay of  $3T_c - t_j - P_j/R_j^u$  and the ONU buffer keeps an amount of  $Q_j - P_j$  bits for the subsequent transmission<sup>1</sup>. Similarly, in the third cycle,  $2(Q_j - P_j)$  bits will remain in the ONU buffer to be transmitted on the fourth cycle. In general, there will be a maximum delay of  $(N + 2)(T_c - t_j - N(P_j/R_j^u))$  in the  $N$ th scheduling cycle, where  $N(Q_j - P_j)$  is accumulated in the ONU buffer. This accumulation continues until  $N_j^C$  cycles after which the buffer of ONU  $j$  is full, and the subsequent generated packets will be lost at the ONU. In other words:

$$N_j^C = \left\lceil \frac{F_j}{Q_j - P_j} \right\rceil \quad (6.9)$$

where  $F_j$  is the buffer size of ONU  $j$ . Therefore, the maximum delay can be derived from (6.8).

### 6.3.3 ILP Model

In [90], the offline scheduling problem in a WDM-PON is formulated as a non-joint optimization problem, where the grant sizing is done in advance using a bandwidth allocation method such as “limited service” or “gated service” [68]. The authors presented an ILP model based on the scheduling theory, where each ONU is considered as a job, its grant size defines the processing time, and the channels used for transmission on the PON represent machines. Therefore, the problem reduces to a “Parallel Machine” (PM) scheduling problem, where a set of jobs, with specific processing times, are executed on a set of machines. Using the same concept, we derive an ILP model for joint scheduling and bandwidth allocation in 10G-EPON coexisting with 1G WDM-PON. Our only channel restriction is that the 10G-ONUs can only

---

<sup>1</sup>We understand that the transmitted data is encapsulated in Ethernet frames in every EPON system and the frames are not allowed to be fragmented. However, throughout this work we assume that ONU grants can be arbitrated at the byte level. This is a reasonable assumption when there are several Ethernet frames with different packet sizes in the requested grant.

be granted on the 10G channel. In our model the size of transmission window for ONU  $j$  on channel  $i$  is a variable which is determined inside the model along with the schedule.

**Sets:**

$O_{1G}$  = set of 1G-ONUs

$O_{10G}$  = set of 10G-ONUs

$O_T$  = set of all existing ONUs ( $O_T = O_{1G} \cup O_{10G}$ )

$\Lambda_{1G}$  = set of WDM 1G channels

$\Lambda_T$  = set of all transmission channels ( $\Lambda_T = \Lambda_{1G} \cup \{\lambda_{10G}\}$ )

**Input Parameters:**

$Q_j$  = requested bandwidth of ONU  $j$

$B_{min}(i, j)$  = minimum guaranteed bandwidth for ONU  $j$  on channel  $i$

$\Delta_j^{max}$  = maximum affordable delay per ONU  $j$

$F_j$  = buffer size of ONU  $j$

$R_j^T$  = transmission line rate for ONU  $j$  (1Gb/s for 1G-ONUs and 10Gb/s for 10G-ONUs)

$$\delta_{ij} = \begin{cases} 1 & \text{if channel } i \text{ is supported by ONU } j \\ 0 & \text{otherwise.} \end{cases}$$

**Output Variables:**

$C_{max}$  = maximum completion time of all channels or the makespan

$t_{ij}$  = length of transmission window for ONU  $j$  on channel  $i$

$$x_{ikj} = \begin{cases} 1 & \text{if position } k \text{ on channel } i \text{ is selected for ONU } j; \\ 0 & \text{otherwise.} \end{cases}$$

$$y_{ikj} = \begin{cases} t_{ij} & \text{if } x_{ikj} = 1; \\ 0 & \text{otherwise.} \end{cases}$$

$$\alpha_{ij} = \begin{cases} 1 & \text{if channel } i \text{ is assigned to ONU } j; \\ 0 & \text{otherwise.} \end{cases}$$

Our objective is to minimize the maximum completion time:

$$\min C_{max} \quad (6.10)$$

The model should guarantee that the makespan is not less than the completion time on any wavelength ( $C_i$ ) which can be determined as:

$$C_i = \sum_{j \in O_T} \sum_{k \in O_T} k \delta_{ij} t_{ij} x_{ikj} \quad (6.11)$$

In order to avoid quadratic terms in our model, we define variables  $y_{ikj} = t_{ij} \times x_{ikj}$ . Using these variables, we write constraint (6.12) to assure that the makespan is greater than the completion time of each channel.

$$C_{max} \geq \sum_{j \in O_T} \sum_{k \in O_T} k \delta_{ij} y_{ikj} \quad i \in \Lambda_T \quad (6.12)$$

Variables  $y_{ikj}$  are determined in the following set of linearization constraints where  $L$  is a large positive number:

$$y_{ikj} \leq t_{ij} + L(1 - x_{ikj}) \quad i \in \Lambda_T, k, j \in O_T; \delta_{ij} = 1 \quad (6.13)$$

$$y_{ikj} \geq t_{ij} - L(1 - x_{ikj}) \quad i \in \Lambda_T, k, j \in O_T; \delta_{ij} = 1 \quad (6.14)$$

$$y_{ikj} \leq L \times x_{ikj} \quad i \in \Lambda_T, k, j \in O_T; \delta_{ij} = 1 \quad (6.15)$$

Constraints (6.16) and (6.17) determine the channel and time slot assignment of upstream bandwidth request of each ONU based on the parallel machine model presented in [90]. Constraint (6.16) ensures that each ONU is assigned to only one

scheduling position and constraint (6.17) guarantees that each scheduling position is assigned to no more than one ONU.

$$\sum_{i \in \Lambda_T} \sum_{k \in O_T} x_{ikj} = 1, \quad j \in O_T \quad (6.16)$$

$$\sum_{j \in O_T} x_{ikj} \leq 1, \quad i, k \in \Lambda_T. \quad (6.17)$$

Constraint (6.18) indicates that on each scheduling round, only one channel is assigned to each ONU. Also, by involving parameter  $\delta_{ij}$ , this constraint implies that the 10G-ONUs can not be allocated on 1G channels.

$$\alpha_{ij} = \delta_{ij} \sum_{k \in O_T} x_{ikj}, \quad i \in \Lambda_T, j \in O_T \quad (6.18)$$

Constraints (6.19)-(6.21) are required for bandwidth allocation based on the discussion in Section 6.3.1. In these constraints, for all  $i \in \Lambda_T$  and  $j \in O_T$  we have:

$$R_j^T \times t_{ij} = \alpha_{ij} \times Q_j \quad Q_j \leq B_{min}(i, j) \quad (6.19)$$

$$R_j^T \times t_{ij} \leq \alpha_{ij} \times Q_j \quad Q_j > B_{min}(i, j) \quad (6.20)$$

$$R_j^T \times t_{ij} \geq \alpha_{ij} \times B_{min}(i, j) \quad Q_j > B_{min}(i, j) \quad (6.21)$$

Constraint (6.23) is for limiting the maximum packet delay per each ONU. Note that expressions (6.8) and (6.9) include nonlinear terms which can not be involved in our ILP model. Hence, to keep the model linear and to obtain a reasonable delay, we approximate  $N_j^C$  as:

$$N_{ij}^C \cong \left\lceil \frac{F_j}{Q_j - B_{min}(i, j)} \right\rceil \quad (6.22)$$

This approximation is based on the fact that in the joint model, the allocated bandwidth is desired to approach the minimum guaranteed bandwidth as much as possible. After replacing  $N_j^C$  with  $N_{ij}^C$ , we rewrite equation (6.9) in the form of an ILP

constraint as follows:

$$(N_{ij}^C + 2)T_c - t_{ij} - \left( N_{ij}^C \cdot \frac{R_j^T}{R_j^u} \right) t_{ij} \leq L(1 - \alpha_{ij}) + \Delta_j^{max}$$

$$i \in \Lambda_T, j \in O_T; Q_j > B_{min}(i, j) \quad (6.23)$$

where  $T_c$  is the total scheduling length considering the initial gap on each cycle, i.e.,  $T_c = ISCG + C_{max}$ . This constraint implies that the delay for ONU  $j$  is less than a predetermined value  $\Delta_j^{max}$ , when ONU  $j$  is scheduled on  $\lambda_i$ .

We note that this model can be used for the case that the 10G channel is dedicated to 10G-ONUs, as well as the case that it is shared with 1G-ONUs. The required changes are reflected in parameter  $\delta_{ij}$  that determines whether a 1G-ONU can transmit on the 10G channel or not. Accordingly, the value of  $B_{min}(i, j)$  changes depending on whether the 10G channel is shared or dedicated.

## 6.4 A Tabu Search Heuristic for Solving the Scheduling Problem

Clearly, the ILP model developed in section 6.3.3 can only be solved for small network instances, as will be shown later. In addition, we have to make an approximation for the delay expression in (6.8) in order to avoid nonlinearity in the model. Thus, it is vital to develop a heuristic in order to involve the nonlinear terms and obtain near-optimal solutions in a reasonable amount of time. To this end, we develop a Tabu search method for solving the joint scheduling and bandwidth allocation problem. Our Tabu search heuristic starts from an initial feasible solution and iterates using two types of moves for obtaining the neighbor solution. One is reordering and moving the ONU grants from one wavelength to another supported wavelength. In this move, the neighborhood of the current solution is obtained by moving a transmission

window of an ONU from its assigned wavelength to another supported wavelength (if there is any). Note that this move can not be applied for the grants of 10G-ONUs, since they can only transmit on the 10G channel. The other move is reducing the transmission window sizes of ONUs whose bandwidth request is larger than the minimum guaranteed bandwidth of the considered channel. For these ONUs, we chose a grant size in the interval given by inequality (6.6) in a way that the imposed delay in expression (6.8) is minimized.

Using these two moves, our Tabu search algorithm performs a local search to explore new feasible solutions. In each iteration of the procedure, both moves are assessed, and the one that yields the best result is chosen as the final move to be performed. Our Tabu method also makes use of a short term memory (Tabu list) that stores information associated with recently explored solutions in order to avoid cycling. For example, the Tabu list contains the positions of the swapped grants, and any move that schedules an ONU grant back to its old position is considered Tabu (i.e., forbidden). Further, and similar to [104], an aspiration criterion which allows to overwrite the Tabu status of a move is used, so that any move that results to higher improvement is considered regardless of the status of the move. Search diversification is obtained by allowing the algorithm to make restart and random perturbations.

The Tabu search algorithm needs some stopping criteria. One stopping criterion is to iterate for a certain number of iterations depending on the number of ONUs. Furthermore, we note that the utilization of different channels varies as the grants of ONUs are resized and reordered among different channels. Therefore, we consider another stopping criterion such that the algorithm runs until the last scheduled ONUs on all wavelength channels have the same finishing time. This is equivalent to maximizing the average bandwidth utilization (measured as the ratio of the sum of ONU transmission times to the total scheduling length of all channels).

## 6.5 Numerical Results

We evaluate our ILP and heuristic models on various network scenarios based on the network architecture in Figure 6.1. We implemented the ILP models for the joint and non-joint scheduling and bandwidth allocation problem in C++, using the “CPLEX Concert Technology” and solved them using the solver CPLEX 11.0.1 [18]. We also used C++ for the implementation of the Tabu search procedure described in Section 6.4. We consider different groups of experiments, by varying the number of 1G- and 10G-ONUs ( $N_{1G}$ ,  $N_{10G}$ ) as well as the number of available 1G channels ( $M_{1G}$ ). In each group, we conduct various experiments assuming that the bandwidth requirement of each ONU is randomly uniformly distributed over the intervals listed in Table 6.2. For obtaining the minimum guaranteed bandwidth in expressions (6.3) and (6.4), we assume  $n_{1G} = 0.2N_{1G}$ ; that is, 20% of 1G-ONUs are assumed to be scheduled on the 10G channel when it is shared between 1G- and 10G-ONUs. Obviously,  $n_{1G} = 0$  when the 10G channel is dedicated to 10G-ONUs. The rest of the network parameters are listed in Table 6.1.

Table 6.1: Network Parameters

Transmission rate of 1G-ONUs	1 Gbps
Transmission rate of 10G-ONUs	10 Gbps
buffer size of 1G-ONUs	1 Mbytes
buffer size of 10G-ONUs	10 Mbytes
Guard bandwidth between adjacent slots	1.5 $\mu$ s
Inter-Scheduling Cycle Gap (ISCG) of each channel	110 $\mu$ s

Table 6.2: Load distribution for 1G- and 10G-ONUs in different experiments

Experiment	Load of 1G-ONUs	Load of 10G-ONUs
$E_{11}$	$[0.5B_{1G}(\lambda_{1G}), 0.9B_{1G}(\lambda_{1G})]$	$[0.2B_{10G}(\lambda_{10G}), 0.7B_{10G}(\lambda_{10G})]$
$E_{12}$	$[0.5B_{1G}(\lambda_{1G}), 0.9B_{1G}(\lambda_{1G})]$	$[0.7B_{10G}(\lambda_{10G}), 1.2B_{10G}(\lambda_{10G})]$
$E_{21}$	$[0.9B_{1G}(\lambda_{1G}), 1.3B_{1G}(\lambda_{1G})]$	$[0.2B_{10G}(\lambda_{10G}), 0.7B_{10G}(\lambda_{10G})]$
$E_{22}$	$[0.9B_{1G}(\lambda_{1G}), 1.3B_{1G}(\lambda_{1G})]$	$[0.7B_{10G}(\lambda_{10G}), 1.2B_{10G}(\lambda_{10G})]$
$E_{31}$	$[1.3B_{1G}(\lambda_{1G}), 1.7B_{1G}(\lambda_{1G})]$	$[0.2B_{10G}(\lambda_{10G}), 0.7B_{10G}(\lambda_{10G})]$
$E_{32}$	$[1.3B_{1G}(\lambda_{1G}), 1.7B_{1G}(\lambda_{1G})]$	$[0.7B_{10G}(\lambda_{10G}), 1.2B_{10G}(\lambda_{10G})]$

### 6.5.1 Evaluation of Different Methods

In the first set of our experiments, we consider two network scenarios; i.e.,  $N_1$  with 2 10G-ONUs, 8 1G-ONUs, and two 1G WDM channels and  $N_2$  with 4 10G-ONUs, 16 1G-ONUs, and three 1G WDM channels. We compare our joint ILP model and Tabu heuristic with the non-joint ILP model presented in [105] which is a modified version of the model presented in [90]. For the non-joint model, we assume that the bandwidth allocation is carried out in advance using the “gated” grant sizing technique, where the grant size for an ONU is simply the queue size reported by that ONU [68]. Each scheduling method is employed for two cases; i.e.,  $\lambda_{XG}$  is either shared with 1G-ONUs or it is dedicated to 10G-ONUs.

Tables 6.3 presents the obtained makespan of different methods for two network scenarios, when a separate or shared DBA is used; that is,  $\lambda_{XG}$  is dedicated to 10G-ONUs or it is shared with 1G-ONUs. Similarly, the average channel utilization of different methods are listed in Table 6.4 for different scheduling scenarios. The results for the joint ILP model can not be obtained for larger network instances and higher traffic loads, due to high computational time (more than a few hours). First, we observe that the Tabu heuristic outperforms the joint ILP model in most of the cases. This is due to the fact that the Tabu method can effectively deal with the nonlinear term in (6.8), rather than making approximation as in the ILP method, which yields in underestimating the maximum delay. Second, we can clearly recognize the consistent improvement achieved when the 10G channel is shared with 1G-ONUs. As expected, a shared DBA module yields higher channel utilization and shorter scheduling periods, as  $\lambda_{XG}$  is more efficiently utilized by some of the 1G-ONUs.

### 6.5.2 Upgrading ONUs from 1Gbps to 10Gbps

We investigate the performance benefits obtained by the users when they undergo rate upgrade from 1G to 10G. We evaluate the joint Tabu. To this end, we evaluate

Table 6.3: Makespan ( $\mu\text{sec}$ )

Group 1 ( $N_{IG} = 8, N_{XG} = 2, M_{IG} = 2$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	1698	1695	1698	1355	1342	1345
$E_{12}$	1698	1695	1698	1667	1607	1670
$E_{21}$	2498	2003	2475	1881	1715	1873
$E_{22}$	2498	2003	2466	2315	2251	2313
$E_{31}$	3375	2003	3189	2506	2402	2416
$E_{32}$	3375	2003	3183	3105	2578	3044
Group 2 ( $N_{IG} = 16, N_{XG} = 4, M_{IG} = 3$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	1950	1856	1948	1658	1585	1647
$E_{12}$	1953	1856	1956	1816	1687	1780
$E_{21}$	2687	2255	-	2209	1977	-
$E_{22}$	2826	2255	-	2335	1573	-
$E_{31}$	3839	2255	-	-	2779	-
$E_{32}$	3839	2255	-	3191	2779	-

Table 6.4: Average Channel Utilization (percentage).

Group 1 ( $N_{IG} = 8, N_{XG} = 2, M_{IG} = 2$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	73.47	73.46	73.47	90.67	91.32	91.29
$E_{12}$	84.75	84.76	84.75	86.23	89.11	86.09
$E_{21}$	71.04	72.33	71.07	93.05	93.46	92.81
$E_{22}$	78.97	82.13	78.91	84.93	85.23	84.37
$E_{31}$	70.39	73.66	70.64	93.77	95.53	93.96
$E_{32}$	78.82	87.57	80.00	85.44	86.26	83.87
Group 2 ( $N_{IG} = 16, N_{XG} = 4, M_{IG} = 3$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	77.25	78.00	77.22	90.00	91.39	90.38
$E_{12}$	81.97	83.07	81.75	87.80	92.51	89.38
$E_{21}$	77.04	73.27	-	92.92	94.64	-
$E_{22}$	77.02	77.77	-	92.48	90.63	-
$E_{31}$	74.30	69.92	-	-	96.09	-
$E_{32}$	77.03	74.47	-	92.14	96.09	-

the joint Tabu method on a network instance when the 1G- and 10G-ONUs are using the same DBA module. We consider a network consisting of six 1G WDM channels and a total of 40 ONUs. We conduct 5 experiments where the 1G-ONUs are gradually being upgraded to 10G. As can be seen in Table 6.5, increasing the number of 10G-ONUs yields a smaller makespan and shorter maximum expected delays (measured as the ratio of the sum of maximum delay per ONU obtained from expressions (6.7) and (6.8) to the total number of ONUs). The reason is that when an ONU is upgraded to 10G, its requested bandwidth becomes much less than the minimum guaranteed bandwidth of 10G-ONUs which is given by expression (6.2). Therefore, the upgraded ONU would be allocated its whole request on the 10G channel with a 10 times smaller transmission window. However, the average channel utilization decreases, since the request of new upgraded 10G-ONUs can only be transmitted on 10G channel and therefore the utilization of 1G channels decreases. This means that the network becomes under-utilized and its potential bandwidth becomes wasted if the offered load is not increased to catch up with the line upgrade to 10G. In other words, to exploit the new capabilities of the upgraded system, more subscribers should be granted, resulting in higher traffic loads.

Table 6.5: Results of the joint Tabu method for a network with 6 1G WDM channels, and a total of 40 ONUs, when the 10G channel is shared with 1G-ONUs which are gradually upgraded to 10G.

$N_{XG}$	$N_{IG}$	Makespan ( $\mu\text{sec}$ )	Expected Maximum Delay ( $\mu\text{sec}$ )	Average Channel Utilization (%)
8	32	2698	23269	92.03
12	28	2614	20997	90.32
16	24	2562	17404	83.11
20	20	2500	6622	80.97
24	16	2316	4783	73.37

To better illustrate this issue, we conduct a set of experiments where the load of the network is gradually increased while the 1G-ONUs are upgraded to 10G. The obtained results are presented in Figure 6.4. For increasing traffic load, experiment

$E_{22}$  in Table 6.2 is evaluated for each of the network scenarios in Table 6.5. We note that ONU requests are generated on a constant bit rate and the total load of the network can be computed as the summation over the instantaneous bitrates of all ONUs. The overall traffic load (measured in Gb/s) is given in Figure 6.4(a) and the corresponding makespan, maximum delay, and channel utilization are presented in Figure 6.4(b), Figure 6.4(c), and Figure 6.4(d), respectively. We can observe that, by increasing the traffic load from 13.5 Gb/s to 17 Gb/s, there will be a slight increase in the makespan for the upgraded system, while the maximum delay and average channel utilization stay in an acceptable interval.

### 6.5.3 Packet Level Simulation

The efficiency of a scheduling method can be better indicated in a packet level system, where requested traffics are scheduled in real Ethernet frames which are undergone one the discussed scheduling techniques. Regarding this issue, we carry out packet-level simulations to study the performance of the proposed scheduling methods in a tangible packet level system. We simulated the operation of the scheduling methods using OMNet++, a discrete event simulator [19]. We assume that the ONU traffic loads are generated at a constant bit-rate (CBR) based on their instantaneous data rates, and the packet size is uniformly distributed between 64 and 1518 bytes. The numerical results are collected for experiments  $E_{11}$ ,  $E_{21}$ , and  $E_{31}$  in network instances  $N_1$  and  $N_2$  when  $\lambda_{XG}$  is either shared with 1G-ONUs or dedicated to 10G-ONUs. For each experiment, we take the scheduling solutions from non-joint ILP [105], joint Tabu, and joint ILP as inputs to our network simulator. The performance metrics are the average queuing delay and average packet loss, which are presented in Tables 6.6 and 6.7, respectively.

The obtained results are in accordance with our previous discussions. As expected, sharing  $\lambda_{XG}$  with 1G-ONUs can significantly improve the overall performance in

Table 6.6: Average Queuing Delay ( $\mu\text{sec}$ )

Group 1 ( $N_{IG} = 8, N_{XG} = 2, M_{IG} = 2$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	2074	2072	2074	1679	1664	1667
$E_{21}$	56840	47671	55946	15683	13697	13440
$E_{31}$	61450	42621	58785	46176	44347	44798
Group 2 ( $N_{IG} = 16, N_{XG} = 4, M_{IG} = 3$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	41355	42545	41604	41355	48808	42073
$E_{21}$	94385	82513	-	78366	71259	-
$E_{31}$	87730	87783	-	-	86532	-

Table 6.7: Average Packet Loss Rate (percentage)

Group 1 ( $N_{IG} = 8, N_{XG} = 2, M_{IG} = 2$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	0	0	0	0	0	0
$E_{21}$	23.36	8.85	22.16	0.44	0.38	0.27
$E_{31}$	42.56	17.34	40.03	23.60	20.49	21.34
Group 2 ( $N_{IG} = 16, N_{XG} = 4, M_{IG} = 3$ )						
Experiment	Separated DBA			Shared DBA		
	Non-joint ILP	joint Tabu	joint ILP	Non-joint ILP	joint Tabu	joint ILP
$E_{11}$	6.67	7.51	7.01	6.66	11.15	7.10
$E_{21}$	28.95	18.92	-	14.65	6.46	-
$E_{31}$	50.27	50.23	-	-	49.10	-

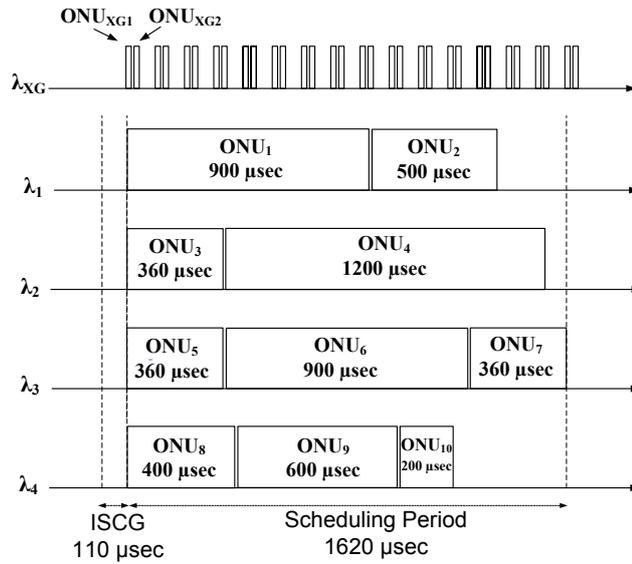
terms of packet loss and average delay. This is due to the fact that, by sharing the 10G-channel, more bandwidth can be allocated to 1G-ONUs in shorter scheduling period. However, we should note that the minimum guaranteed bandwidth of 10G-ONUs decreases as the 10G-channel is shared with 1G-ONUs. Therefore, 10G-ONUs may have larger drop rate and delay for higher traffic loads. Another conclusion is that our Tabu is a promising method for the joint scheduling and grant sizing problem of 1G-WDM and 10G-TDM ONUs; it provides close to optimal solutions with significantly smaller computational costs compared to the ILP models.

In Figure 6.5, we compare our joint Tabu method with the online NASC and

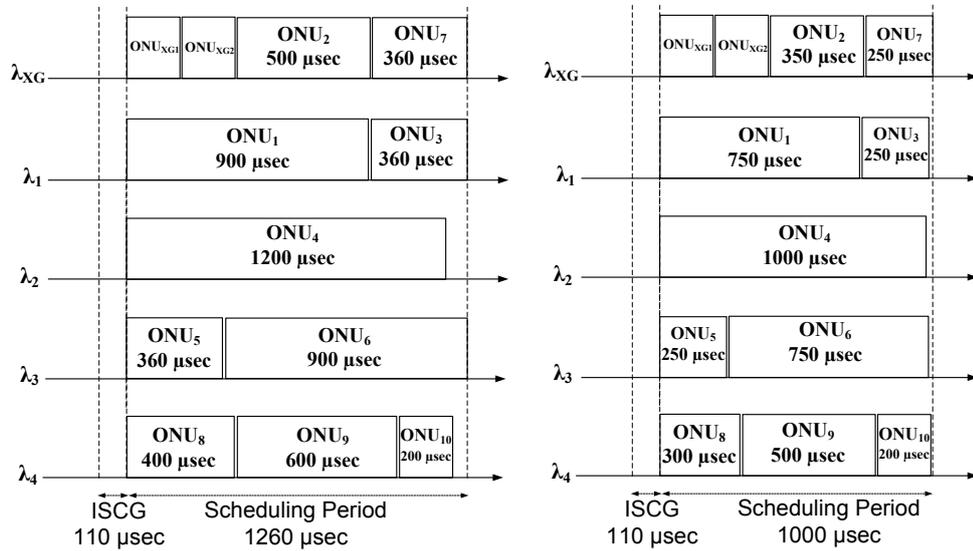
the offline scheduling method presented in [90], that is non-joint scheduling based on minimizing the total completion time (we call this method NJS-TCT). The upper two figures show the average packet loss and the lower two figures show the average queueing delay for two groups of network topologies. It can be recognized that our Tabu method consistently outperforms the offline method of [90] in terms of packet loss and average queueing delay. We also observe that for medium load experiment  $E_{21}$ , our Tabu method exhibits a better performance than the online NASC. It should also be noted that the offline methods (either Tabu or ILP) have always a higher channel utilization than online NASC.

## 6.6 Conclusion

We studied the problem of optimal scheduling and bandwidth allocation in next generation 10G-EPON coexisting with 1G WDM-PONs. We derived an ILP model for offline joint scheduling and bandwidth assignment for providing this coexistence. For large network instances, the size of the ILP model becomes prohibitively large. Therefore, we introduced a Tabu Search heuristic to achieve near optimal solutions in notably shorter computation times. We explored the tradeoff which exists in terms of delay, scheduling length, and channel utilization, when separate or the same DBA and scheduling modules are used for 1G- and 10G-ONUs. We also showed the influence of gradually upgrading the 1G-ONUs to 10G-ONUs in a network with a fixed number of ONUs. We conclude that upgrading WDM 1G-ONUs to TDM 10G-ONUs can improve the quality of service experienced by end users, yet it would decrease the channel utilization on the existing 1G channels.



(a) Separate scheduling modules for 1G- and 10G-EPON



(b) Same scheduling module for 1G- and 10G-EPON

(c) Joint scheduling and bandwidth allocation for 1G- and 10G-EPON

Figure 6.2: An illustrative example for comparing different scheduling frameworks

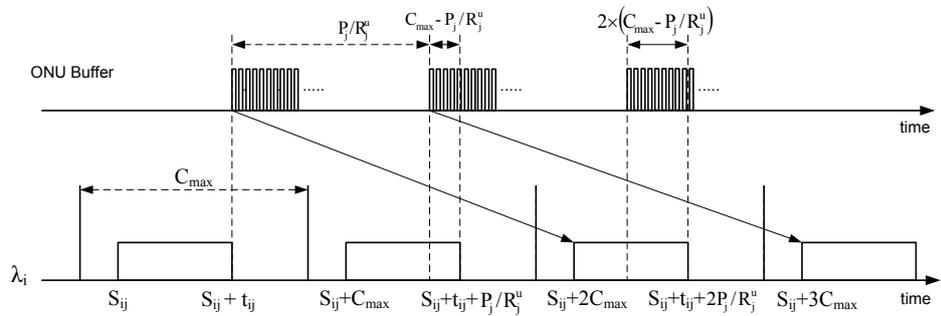


Figure 6.3: Dynamics of bandwidth allocation

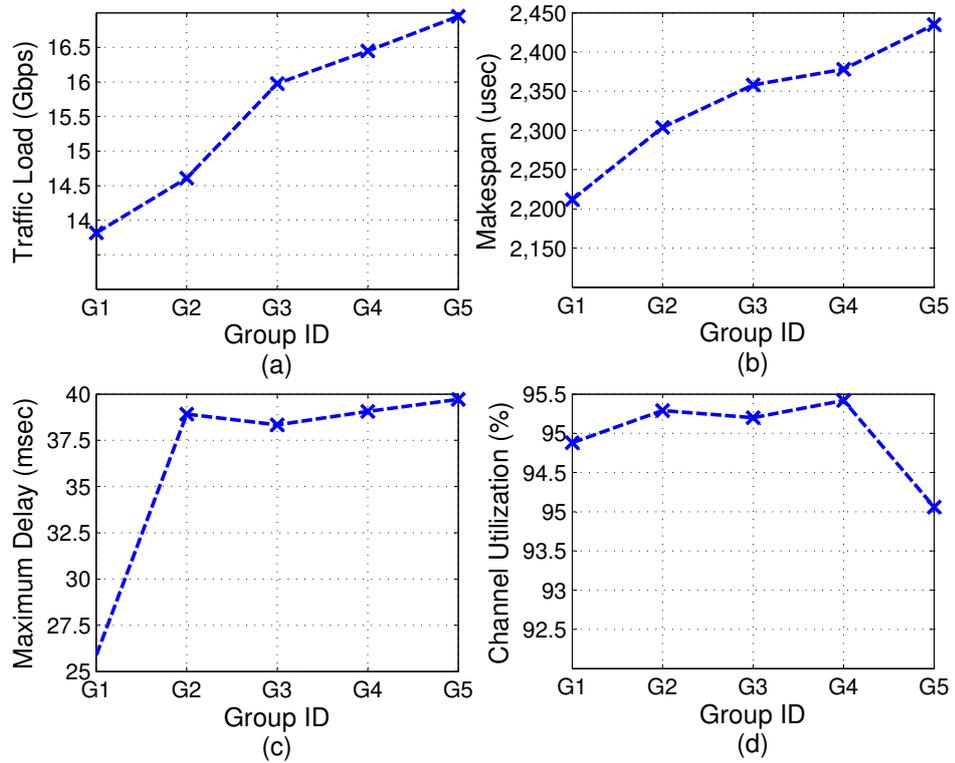


Figure 6.4: Results of upgrading scenarios. (a) network traffic load (Gb/s), (b) makespan ( $\mu\text{sec}$ ), (c) Expected maximum delay (msec), (d) average channel utilization

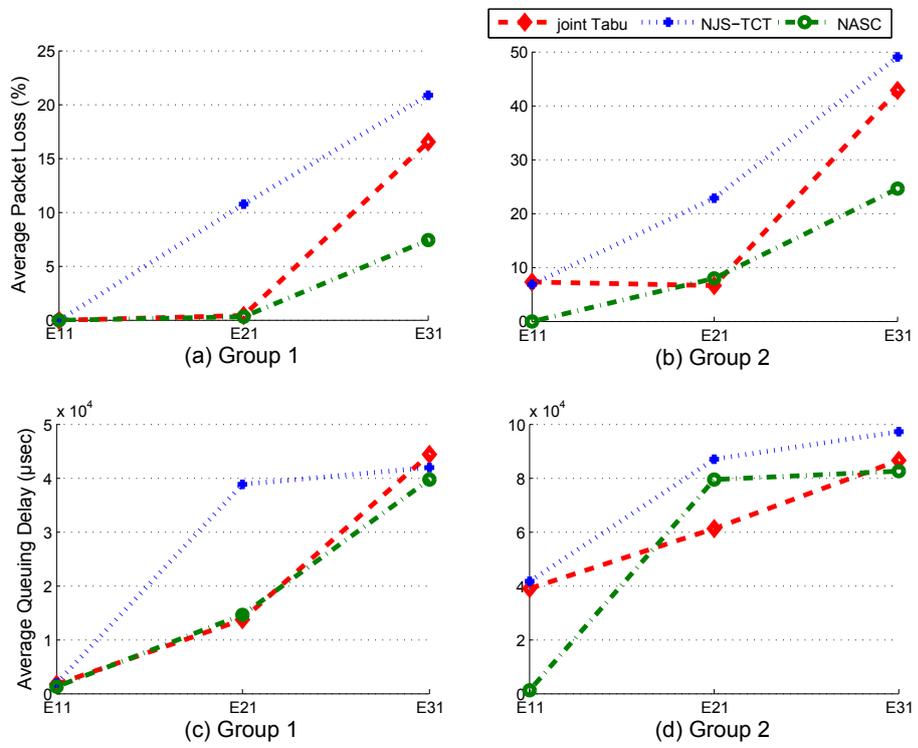


Figure 6.5: Comparison of our Tabu method with online NASC and NJS-TCT

# Chapter 7

## Conclusion and Future Directions

### 7.1 Conclusions

Towards the realization of reliable and efficient optical networks, several issues must be addressed, among which we considered two of the most important problems; survivability in optical long-haul networks and resource management in optical access networks.

To address the survivability problem, first we presented enhanced models for availability-aware provisioning in  $p$ -cycle based networks. Our approach builds upon previous work in [2] and we resolved two main flaws in the prior work to achieve an exact non-joint optimization model for the service provisioning problem. Then, we introduced several techniques to deal with the scalability issue and speed up the process of achieving the solution. Our results indicate that for higher values of the required availability, the effect of resolving the flaws in previous model is more significant and therefore our model requires less capacity investments from network operators in comparison to prior work. We further analyzed the tradeoffs between reducing the mean time to repair (MTTR) and deploying more protection capacity to achieve a certain level of service availability. This study helps network operators wisely allocate their budget to maintain a certain level of service guarantees in their

network.

Next, we investigated the survivable network design problem based on pre-cross-connected protection trails, known as  $p$ -trails. By taking the sharing capability of  $p$ -trails into account, we introduce optimization models to verify the efficiency of  $p$ -trails. We derived two ILP models for survivable network design with single failure restorability using  $p$ -trails. As the size of the network increase, the number of candidate trails grows exponentially and therefore the size of our first ILP and its computation time become excessive and prohibit us from obtaining solutions for practical networks. To overcome the scalability issue, we derived a second model based on the “Column Generation” (CG) decomposition technique, and we showed that our CG model is a remarkably scalable ILP model for  $p$ -trail based network design, yielding to an optimal solution with less required spare capacity compared to the  $p$ -cycle solution.

Our study on the problem of resource management in optical address networks is twofold. First, we studied the problem of grant scheduling and bandwidth allocation in evolutionarily upgraded WDM PONs. We presented three new ILP models for the non-joint and joint scheduling and grant sizing problem. Since the joint ILP models are very hard to solve, except for small network instances, we introduced Tabu search heuristic for achieving near-optimal solutions. Our experiments show that the joint scheduling and sizing algorithms outperform the non-joint models in terms of scheduling cycle length. We also derived a new model for the non-joint problem that outperform the previous non-joint model in terms of makespan, utilization, queuing delay and packet loss. Furthermore, it was observed that the introduced Tabu search heuristic is a promising solution for the joint scheduling and grant sizing problem. While significantly reducing the computation time compared to the sequential and joint ILP models, our Tabu heuristic provides close to optimal solutions. Second, we addressed the problem of optimal scheduling and bandwidth allocation in

next generation 10G-EPON coexisting with 1G-WDM PONs. We derived an ILP model for offline joint scheduling and bandwidth assignment to provide coexistence. For large network instances, the size of the ILP model becomes prohibitively large. Therefore, we introduced another Tabu search heuristic to achieve near optimal solutions in notably shorter computation times. We explored the tradeoff which exists in terms of delay, scheduling length, and channel utilization, when separate or the same dynamic bandwidth allocation (DBA) and scheduling modules are used for 1G- and 10G-ONUs. We also showed the impact of gradually upgrading 1G-ONUs to 10G-ONUs on a network with a fixed number of ONUs. This study declares that upgrading WDM 1G-ONUs to TDM 10G-ONUs improve the quality of service experienced by end-users, but it would decrease the channel utilization on the existing 1G channels. In other words, to exploit the new capabilities of the upgraded system, more subscribers should be granted, resulting in higher traffic loads.

## 7.2 Future Work

The work presented in this thesis provided considerable performance enhancements of optical long-haul and access networks. However, there remain several future research directions that may provide additional benefits.

The availability models presented in Chapter 3 can be extended to path-protecting  $p$ -cycles. In other words, the availability-aware design model presented in [67] for FIPP should be revisited and elaborated to resolve the probable shortcomings. Regarding that FIPP generally outperforms link-based  $p$ -cycles, providing a more accurate availability-aware model for FIPP can result in significant improvement of service availability of network demands. Furthermore, the  $p$ -trail design model presented in Chapter 4 could be enhanced to address the availability requirements. Specifically, the protection domain of each  $p$ -trail should be defined and the dual failure scenarios resulting in service outage must be considered in a  $p$ -trail based mesh network.

This availability analysis of  $p$ -trail networks will be by far more complicated than that of  $p$ -cycle based network. Therefore, several techniques will be needed to address the scalability issue. In addition,  $p$ -trails can be extended to protect the whole path rather than individual links. Designing path-based  $p$ -trails and analyzing their availability is another interesting future direction.

The study on resource management in optical access network can be enhanced to consider next-generation long-reach PONs (LR-PONs). LR-PONs are poised to be the next step in the evolution of access-metro optical networks. They essentially have the same topology as PONs and are characterized by a longer distance between the optical line terminal (OLT) and the optical network units (ONUs) as well as a larger number of ONUs. Hence, although both PON and LR-PON use one upstream and one downstream channel, the maximal reach of standardized PONs is 20km whereas LR-PONs are expected to span lengths of up to 100km. Moreover, LR-PONs are expected to operate at a line rate of  $R_d = 10\text{Gb/s}$  and to have 2000 to 4000 ONUs. The shift from PONs to LR-PONs translates into longer propagation times and round-trip times (RTTs) from the OLT to the ONUs. This stipulates more sophisticated and efficient grant scheduling methods while taking care of the imposed transmission delay. In recent years, some efforts have been made to address the dynamic bandwidth allocation problem in LR-PONs [106–109]. Yet, no research has been reported on investigating the effect of different polling and system parameters on frame delay in LR-PON. Designing a framework for quantifying transmission delay and exploring the roles played by various system parameters in a LR-PON represents an exciting future research direction.

# Bibliography

- [1] B. Wu and P.-H. Ho, “ILP formulations for non-simple  $p$ -cycle and  $p$ -trail design in WDM mesh networks,” *Computer Networks*, vol. 54, no. 5, pp. 716–725, Apr. 2010.
- [2] D. S. Mukherjee, C. Assi, and A. Agarwal, “An alternative approach for enhanced availability analysis and design methods in  $p$ -cycle-based networks,” *IEEE JSAC*, vol. 24, no. 12, pp. 23–34, Dec. 2006.
- [3] W. D. Grover, *Mesh-Based Survivable Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking*. Prentice-Hall, Upper Saddle River, New Jersey, 2004.
- [4] R. Ramaswami, “Optical networking technologies: What worked and what didnt,” *IEEE Communications Magazine*, vol. 44, no. 9, pp. 132–139, Sept. 2006.
- [5] R. Ramaswami and K. Sivarajan, *Optical Networks: A Practical Perspective*, 2nd ed. Morgan Kaufmann Publishers, San Francisco, CA, USA, 2002.
- [6] T. H. Wu, D. J. Kolar, and R. H. Cardwell, “Survivable network architectures for broad-band fiber optic networks: model and performance comparison,” *IEEE/OSA Journal of Lightwave Technology*, vol. 6, no. 11, pp. 1698–1709, Nov. 1988.

- [7] J. Sosnosky and T. H. Wu, "SONET ring applications for survivable fiber loop networks," *IEEE Communications Magazine*, vol. 29, no. 6, pp. 51–58, June 1991.
- [8] T. H. Wu, *Fiber Network Service Survivability*. Artech House, 1992.
- [9] M. Clouqueur and W. D. Grover, "Availability analysis for span-restorable mesh networks," *IEEE JSAC*, vol. 20, pp. 810–821, May 2002.
- [10] W. D. Grover and D. Stamatelakis, "Cycle-oriented distributed preconfiguration: Ring-like speed with mesh-like capacity for self-planning network restoration," in *Proc., IEEE International Conference on Communications (ICC'98)*, Atlanta, GA, USA, June 1998, pp. 537–543.
- [11] M. H. MacGregor, W. D. Grover, and K. Ryhorchuk, "Optimal spare capacity preconfiguration for faster restoration of mesh networks," *Journal of Network and Systems Management*, vol. 5, no. 2, pp. 159–171, June 1997.
- [12] A. Kodian and W. D. Grover, "Failure-independent path-protecting  $p$ -cycles: efficient and simple fully preconnected optical-path protection," *IEEE JLT*, vol. 23, no. 10, pp. 3241–3259, Oct. 2005.
- [13] T. Y. Chow, F. Chudak, and A. M. Ffrench, "Fast optical layer mesh protection using pre-cross-connected trails," *IEEE/ACM Transactions on Networking*, vol. 12, no. 3, pp. 539–548, June 2004.
- [14] D. P. Shea and J. E. Mitchell, "Long-Reach Optical Access Technologies," *IEEE Network*, vol. 21, no. 5, pp. 5–11, Sept./Oct. 2007.
- [15] G. Kramer, B. Mukherjee, and G. Pesavento, "Ethernet PON (ePON): Design and analysis of an optical access network," *Photonic Network Communications*, vol. 3, pp. 307–319, 2001.

- [16] M. P. McGarry, “An evolutionary wavelength division multiplexing upgrade for ethernet passive optical networks,” Master’s thesis, Arizona State University, Spring 2004.
- [17] K. Tanaka, A. Agata, and Y. Horiuchi, “IEEE 802.3av 10G-EPON standardization and its research and development status,” *IEEE/OSA Journal of Lightwave Technology*, vol. 28, no. 4, pp. 651–661, Feb. 2010.
- [18] *ILOG CPLEX 11.0 User’s Manual*, IBM ILOG, Sept. 2007.
- [19] OMNet++, <http://www.omnetpp.org/>, last accessed in Dec. 2010.
- [20] D. Zhou and S. Subramaniam, “Survivability in optical networks,” *IEEE Network*, vol. 14, no. 6, pp. 16–23, Nov./Dec. 2000.
- [21] W. He and A. K. Somani, “Comparison of protection mechanisms: Capacity efficiency and recovery time,” in *Proc., IEEE International Conference on Communications (ICC’07)*, Glasgow, UK, June 2007.
- [22] S. Ramamurthy and B. Mukherjee, “Survivable WDM mesh networks, part II restoration,” in *Proc., IEEE International Conference on Communications (ICC’99)*, Vancouver, British Columbia, Canada, June 1999, pp. 2023–2030.
- [23] W. D. Grover and D. Stamatelakis, “Bridging the ring-mesh dichotomy with  $p$ -cycles,” in *Proc., International Workshop on Design of Reliable Communication Networks (DRCN)*, Munich, Germany, Apr. 2000, pp. 92–104.
- [24] D. Stamatelakis and W. D. Grover, “Rapid span or node restoration in ip networks using virtual protection cycles,” in *Proc., The Canadian Conference on Broadband Research (CCBR ’99)*, Ottawa, Canada, Nov. 1999.

- [25] G. Shen and W. D. Grover, "Extending the  $p$ -cycle concept to path segment protection for span and node failure recovery," *IEEE JSAC*, vol. 21, no. 8, pp. 1306–1319, Oct. 2003.
- [26] P.-H. Ho and H. T. Mouftah, "A novel survivable algorithm for segment shared protection in mesh wdm networks with partial wavelength conversion," *IEEE JSAC*, vol. 22, pp. 1539–1548, Oct. 2004.
- [27] J. Tapolcai, P.-H. Ho, D. Verchre, T. Cinkler, and A. Haque, "A new shared segment protection method for survivable networks with guaranteed recovery time," *IEEE Transactions on Reliability*, vol. 57, no. 2, pp. 272–282, June 2008.
- [28] S. Kini, M. Kodialam, T. V. Laksham, S. Sengupta, and C. Villamizar, "Shared backup label switched path restoration," *Internet Engineering Task Force [Online]*. Available: <http://tools.ietf.org/html/draft-kini-restoration-shared-backup-00.txt>, May 2001.
- [29] R. R. Iraschko, M. MacGregor, and W. D. Grover, "Optimal capacity placement for path restoration in STM or ATM mesh survivable networks," *IEEE/ACM Transactions on Networking*, vol. 6, no. 3, p. 325336, June 1998.
- [30] W. D. Grover and A. Kodian, "Failure-independent path protection with  $p$ -cycles: Efficient, fast and simple protection for transparent optical networks," in *Proc., International Conference on Transparent Optical Networks (ICTON 2005)*, Barcelona, Spain, July 2005, pp. 363–369.
- [31] D. Stamatelakis and W. D. Grover, "Theoretical underpinnings for the efficiency of restorable networks using preconfigured cycles (" $p$ -cycles")," *IEEE Transactions on Communications*, vol. 48, no. 8, pp. 1262–1265, Aug. 2000.
- [32] W. D. Grover and J. Doucette, "Advances in optical network design with  $p$ -cycles: Joint optimization and pre-selection of candidate  $p$ -cycles," in *Proc.*,

*IEEE Lasers and Electro-Optics Society (LEOS) Summer Topicals*, Québec, Canada, July 2002.

- [33] J. Doucette, D. He, W. D. Grover, and O. Yang, “Algorithmic approaches for efficient enumeration of candidate  $p$ -cycles and capacitated  $p$ -cycle network design,” in *Proc., International Workshop on Design of Reliable Communication Networks (DRCN)*, Alberta, Canada, Oct. 2003, pp. 212–220.
- [34] D. Schupke, C. Gruber, and A. Autenrieth, “Optimal configuration of  $p$ -cycles in WDM networks,” in *Proc., IEEE International Conference on Communications (ICC’02)*, New York City, NY, USA, Apr./May 2002.
- [35] W.-D. Zhong and Z. Zhang, “ $p$ -Cycles-based dynamic protection provisioning in optical WDM networks,” *IEICE Transactions on Communications*, vol. E88-B, no. 5, pp. 1921–1926, May 2005.
- [36] D. Stamatelakis, “Theory and algorithms for preconfiguration of spare capacity in mesh restorable networks,” Master’s thesis, University of Alberta, Spring 1997.
- [37] R. Iraschko, M. H. MacGregor, and W. D. Grover, “Optimal capacity placement for path restoration in mesh survivable networks,” in *Proc., IEEE International Conference on Communications (ICC’96)*, Dallas, Texas, USA, June 1996, pp. 1568–1574.
- [38] D. Rajan and A. Atamturk, “Survivable network design: routing of flows and slacks,” in *Telecommunications Network Design and Management, Sixth INFORMS Telecommunications Conference*, 2003, pp. 65–81.
- [39] W. Zhong, F. Zhang, and Y. Jin, “Optimized designs of  $p$ -cycles for survivable multicast sessions in optical WDM networks,” in *International Conference*

*on Communications and Networking in China (CHINACOM '07)*, Shanghai, China, Aug. 2007, pp. 473–478.

- [40] F. Zhang and W. Zhong, “Applying  $p$ -cycles in dynamic provisioning of survivable multicast sessions in optical WDM networks,” in *Proc., Optical Fiber Communication Conference (OFC/NFOEC)*, Anaheim, CA, USA, Mar. 2007.
- [41] T. Feng, L. Ruan, and W. Zhang, “Intelligent  $p$ -cycle protection for multicast sessions in WDM networks,” in *Proc., IEEE International Conference on Communications (ICC '08)*, Beijing, China, May 2008.
- [42] F. Zhang and W. Zhong, “Performance evaluation of  $p$ -cycle based protection methods for provisioning of dynamic multicast sessions in mesh WDM networks,” *Photonic Network Communications Journal*, vol. 16, no. 2, pp. 127–138, 2008.
- [43] B. Wu and P.-H. Ho, “ILP formulations for  $p$ -cycle design without candidate cycle enumeration,” *IEEE/ACM Transactions on Networking*, vol. 18, no. 1, pp. 284–295, Feb. 2010.
- [44] H. Zhang and O. Yang, “Finding protection cycles in DWDM networks,” in *Proc., IEEE International Conference on Communications (ICC'02)*, New York City, NY, USA, May 2002.
- [45] C. Gruber, “Resilient networks with non-simple  $p$ -cycles,” in *Proc., International Conference on Telecommunications (ICT 2003)*, Tahiti, Papeete, French Polynesia, Feb./Mar. 2003, pp. 1027–1032.
- [46] M. E. Lubbecke and J. Desrosiers, “Selected topics in column generation,” *Operations Research*, vol. 53, no. 6, pp. 1007–1023, Nov./Dec. 2005.

- [47] C. Barnhart, E. Johnson, G. Nemhauser, M. Savelsbergh, and P. Vance, “Branch-and-price: Column generation for solving huge integer programs,” *Operations Research*, vol. 46, no. 3, p. 316–329, May/June 1998.
- [48] F. Vanderbeck, “Decomposition and column generation for integer programs,” Ph.D. dissertation, Université Catholique de Louvain, Louvain-la-Neuve, Belgium, 1994.
- [49] V. Chvatal, *Linear Programming*. Freeman, 1983.
- [50] T. Thomadsen and T. Stidsen, “Joint optimization of working and  $p$ -cycle protection capacity,” in *Informatics and Mathematical Modelling*, Technical University of Denmark, DTU, 2004.
- [51] B. Jaumard, C. Rocha, D. Baloukov, and W. D. Grover, “A column generation approach for design of networks using path-protecting  $p$ -cycles,” in *Proc., International Workshop on Design of Reliable Communication Networks (DRCN)*, La Rochelle, France, Oct. 2007.
- [52] C. Rocha and B. Jaumard, “Revisiting  $p$ -cycles / FIPP  $p$ -cycles vs. shared link / path protection,” in *Proc., IEEE International Conference on Computer Communications and Networks (ICCCN 2008)*, Virgin Islands, USA, Aug. 2008.
- [53] —, “A unified framework for shared protection schemes in optical mesh network,” *Pesquisa Operacional*, vol. 29, no. 3, pp. 533–546, Sept./Dec. 2009.
- [54] S. Sebbah and B. Jaumard, “Survivable WDM networks design with non-simple  $p$ -cycle based PWCE,” in *Proc., IEEE Global Communications Conference (GLOBECOM '08)*, New Orleans, LO, USA, Nov./Dec. 2008.
- [55] M. A. H. Clouqueur, “Availability of service in mesh-restorable transport networks,” Ph.D. dissertation, University of Alberta, Canada, 2004.

- [56] M. Clouqueur and W. D. Grover, “Mesh-restorable networks with enhanced dual-failure restorability properties,” *Photonic Network Communications*, vol. 9, no. 1, pp. 7–18, Jan. 2005.
- [57] D. A. Schupke, W. D. Grover, and M. Clouqueur, “Strategies for enhanced dual failure restorability with static or reconfigurable  $p$ -cycle networks,” in *Proc., IEEE International Conference on Communications (ICC'04)*, Paris, France, June 2004.
- [58] D. Leung, S. Arakawa, M. Murata, and W. D. Grover, “Re-optimization strategies to maximize traffic-carrying readiness in WDM survivable mesh networks,” in *Proc., Optical Fiber Communication Conference (OFC/NFOEC)*, Anaheim, USA, Mar. 2005.
- [59] D. A. Schupke, “The tradeoff between the number of deployed  $p$ -cycles and the survivability to dual fiber duct failures,” in *Proc., IEEE International Conference on Communications (ICC'03)*, Anchorage, AK, USA, May 2003.
- [60] —, “Multiple failure survivability in WDM networks with  $p$ -cycles,” in *Proc., IEEE International Symposium on Circuits and Systems (ISCAS)*, Bangkok, Thailand, May 2003, invited Paper.
- [61] H. Wang and H. Mouftah, “ $p$ -cycles in multi-failure network survivability,” in *Proc., International Conference on Transparent Optical Networks*, Barcelona, Spain, July 2005, pp. 381–384.
- [62] C. Gruber and D. Schupke, “Capacity-efficient planning of resilient networks with  $p$ -cycles,” in *Proc. Networks, The International Telecommunication Strategy and Planning Symposium*, 2002.

- [63] D. S. Mukherjee, C. Assi, and A. Agarwal, "Alternate strategies for dual failure restoration using  $p$ -cycles," in *Proc., IEEE International Conference on Communications (ICC'06)*, vol. 6, Istanbul, Turkey, June 2006, pp. 2477–2482.
- [64] A. Kodian and W. Grover, "Multiple-quality of protection classes including dual-failure survivable services in  $p$ -cycle networks," in *Proc. International Conference on Broadband Networks*, vol. 1, Boston, Massachusetts, USA, Oct. 2005, pp. 231–240.
- [65] W. Grover and A. Sack, "High availability survivable networks: When is reducing MTTR better than adding protection capacity?" in *Proc., International Workshop on the Design of Reliable Communication Networks (DRCN)*, vol. 1, France, Oct. 2007, pp. 231–240.
- [66] M. Clouqueur and W. D. Grover, "Availability analysis and enhanced availability design in  $p$ -cycle-based networks," *Photonic Network Communications, Springer Science*, vol. 10, no. 1, pp. 55–71, 2005.
- [67] A. Ranjbar and C. Assi, "Availability-aware design in FIPP  $p$ -cycles protected mesh networks," in *Proc., International Conference on Optical Networking Design and Modeling (ONDM 2008)*, Vilanova i la Geltrú, Spain, March 2008.
- [68] M. P. McGarry, M. Reisslein, and M. Maier, "Ethernet passive optical network architectures and dynamic bandwidth allocation algorithms," *IEEE Communications Surveys and Tutorials*, vol. 10, no. 3, pp. 46–60, 3rd Quarter 2008.
- [69] G. Kramer, B. Mukherjee, and A. Maislos, *Multiprotocol over DWDM: Building the Next Generation Optical Internet*. John Wiley & Sons, Inc., 2003.
- [70] P. E. Green, *Fiber to the Home: The New Empowerment*. John Wiley & Sons, Inc., 2006.

- [71] T. Koonen, “Fiber to the home/fiber to the premises: What, where, and when,” *Proceedings of the IEEE*, vol. 94, no. 5, pp. 911–934, May 2006.
- [72] I. V. de Voorde and G. V. der Plas, “Full Service Optical Access Networks: ATM Transport on Passive Optical Networks,” *IEEE Communications Magazine*, vol. 4, no. 35, pp. 70–75, 1997.
- [73] “ITU-T G.983.1 - Broadband Passive Optical Networks (BPON): General Characteristics,” June 1999.
- [74] M. Hajduczenia, H. J. A. da Silva, and P. P. Monteiro, “EPON versus APON and GPON: a detailed performance comparison,” *Journal of Optical Networking*, vol. 4, no. 5, pp. 298–319, April 2006.
- [75] *Multi-Point Control Protocol (MPCP)*, IEEE 802.3ah task force home page [Online], 2001, <http://www.ieee802.org/3/efm>. [Online]. Available: <http://www.ieee802.org/3/efm>
- [76] J. Zhang and N. Ansari, “Toward Energy-Efficient 1G-EPON and 10G-EPON with Sleep-Aware MAC Control and Scheduling,” *IEEE Communications Magazine*, vol. 49, no. 2, pp. s33–s38, Feb. 2011.
- [77] G. Kramer, “10G-EPON: Drivers, challenges, and solutions,” in *Proc., European Conference on Optical Communication (ECOC '09)*, Vienna, Austria, Sept. 2009, pp. 1–3.
- [78] M. Hajduczenia, H. J. A. D. Silva, and P. Monteiro, “Development of 10 Gb/s EPON in IEEE 802.3av,” *IEEE Communications Magazine*, vol. 46, no. 7, pp. 40–47, July 2008.
- [79] J. Zhang and N. Ansari, “On the Capacity of WDM Passive Optical Networks,” *IEEE Transactions on Communications*, vol. 59, no. 2, pp. 552–559, Feb. 2011.

- [80] K. Grobe and J. P. Elbers, "PON in adolescence: From TDMA to WDM-PON," *IEEE Communications Magazine*, vol. 46, no. 1, pp. 26–34, Jan. 2008.
- [81] M. Zirngibl, C. H. Joyner, L. Stulz, C. Dragone, H. M. Presby, and I. P. Kaminow, "LARnet, a local access router network," *IEEE Photonics Technology Letters*, vol. 7, no. 2, pp. 215–217, Feb. 1995.
- [82] N. J. Frigo, P. D. Magill, T. E. Darcie, P. P. Iannone, M. M. Downs, B. N. Desai, U. Koren, T. L. Koch, C. Dragone, and H. M. Presby, "RITE-Net: A passive optical network architecture based on the remote interrogation of terminal equipment," in *Optical Fiber Communication Technical Digest*, San Jose, CA, USA, Feb. 1994, pp. 43–45.
- [83] F.-T. An, K. S. Kim, D. Gutierrez, S. Yam, S.-T. Hu, K. Shrikhande, and L. G. Kazovsky, "SUCCESS: A next-generation hybrid WDM/TDM optical access network architecture," *IEEE/OSA Journal of Lightwave Technology*, vol. 22, no. 11, pp. 2557–2569, Nov. 2004.
- [84] J.-I. Kani, "Enabling Technologies for Future Scalable and Flexible WDM-PON and WDM/TDM-PON Systems," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 16, no. 5, pp. 1290–1297, Sept./Oct. 2010.
- [85] M. D. Feuer, J. M. Wiesenfeld, J. S. Perino, C. A. Bums, G. Raybon, S. C. Shunk, and N. K. Dutta, "Single-port Laser-amplifier Modulators for Local Access," *IEEE Photonics Technology Letters*, vol. 8, no. 9, pp. 1175–1177, Sept. 1996.
- [86] M. Fujiwara and J. Kani, "Impact of backreflection on upstream transmission in WDM single-fiber loopback access networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 24, no. 9, pp. 740–746, Feb. 2006.

- [87] G. Kramer, B. Mukherjee, and G. Pesavento, "IPACT: A dynamic protocol for an ethernet PON (EPON)," *IEEE Communications Magazine*, vol. 40, no. 2, pp. 74–80, Feb. 2002.
- [88] K. Kwong, D. Harle, and A. Andonovic, "Dynamic bandwidth allocation algorithm for differentiated services over WDM EPONs," in *Proc., International Conference on Communications Systems*, Singapore, Sept. 2004, pp. 116–120.
- [89] F. Clarke, S. Sarkar, and B. Mukherjee, "Simultaneous and interleaved polling: An upstream protocol for WDM-PON," in *Proc., Optical Fiber Communication Conference (OFC/NFOEC)*, Anaheim, CA, USA, Mar. 2006, p. 3.
- [90] M. P. McGarry, M. Reisslein, M. Maier, and A. Keha, "Bandwidth management for WDM EPONs," *Journal of Optical Networking*, vol. 5, no. 9, pp. 637–654, Sept. 2006.
- [91] M. P. McGarry, M. Reisslein, C. J. Colbourn, M. Maier, F. Aurzada, and M. Scheutzow, "Just-in-time scheduling for multichannel EPONs," *IEEE/OSA Journal of Lightwave Technology*, vol. 26, no. 10, pp. 1204–1216, March 2008.
- [92] A. Dhaini, C. Assi, M. Maier, and A. Shami, "Dynamic wavelength and bandwidth allocation in hybrid TDM/WDM EPON networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 25, no. 1, pp. 277–286, January 2007.
- [93] M. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.
- [94] M. Gagnaire and M. Koubaa, "A new control plane for next-generation WDM-PON access systems," in *Proc., IEEE/ICST International Conference on Access Networks (AccessNets)*, Ottawa, Canada, Aug. 2007, pp. 116–120.

- [95] L. Meng, J. El-Najjar, and C. Assi, “A joint transmission grant scheduling and wavelength assignment in multichannel SG-EPON,” *IEEE/OSA Journal of Lightwave Technology*, vol. 27, no. 21, pp. 4781–1492, Nov. 2010.
- [96] M. Maier, M. Herzog, and M. Reisslein, “STARGATE: The Next Evolutionary Step toward Unleashing the Potential of WDM EPONs,” *IEEE Communications Magazine*, vol. 45, no. 5, pp. 50–56, May 2007.
- [97] K. Kanonakis and I. Tomkos, “Improving the Efficiency of Online Upstream Scheduling and Wavelength Assignment in Hybrid WDM/TDMA EPON Networks,” *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 6, pp. 838–848, August 2010.
- [98] E. Mannie and D. Papadimitriou, “Recovery (Protection and Restoration) Terminology for Generalized Multi-Protocol Label Switching (GMPLS),” *Internet Engineering Task Force [Online]*. Available: <http://www.ietf.org/rfc/rfc4427.txt>, March 2006.
- [99] B. Wu, P.-H. Ho, L. Yeung, J. Tapolcai, and H. T. Mouftah, “CFP: Cooperative Fast Protection,” in *Proc., IEEE International Conference on Computer Communications (INFOCOM '09)*, Rio de Janeiro, Brazil, Apr. 2009.
- [100] M. To and P. Neusy, “Unavailability analysis of long-haul networks,” *IEEE JSAC*, vol. 12, no. 1, pp. 100–109, Jan. 1994.
- [101] S. Sebbah and B. Jaumard, “Differentiated quality of service in survivable WDM mesh networks,” in *Proc., IEEE Global Communications Conference (GLOBECOM '09)*, Honolulu, HI, USA, Nov./Dec. 2009.
- [102] M. P. McGarry, M. Maier, and M. Reisslein, “WDM Ethernet Passive Optical Networks,” *IEEE Communications Magazine*, vol. 44, no. 2, pp. S18–S25, Feb. 2006.

- [103] U. Bilge, F. K-rac, M. Kurtulan, and P. Pekgun, “A Tabu Search Algorithm for Parallel Machine Total Tardiness Problem,” *Elsevier Computers and Operations Research*, vol. 31, no. 3, p. 397414, March 2004.
- [104] D. Alcaide, J. Sicilia, and D. Vigo, “A Tabu Search Algorithm for the Open Shop Problem,” *Sociedad de Estadística e Investigación Operativa*, vol. 5, no. 2, pp. 283–296, 1997.
- [105] M. S. Kiaei, L. Meng, C. Assi, and M. Maier, “Efficient scheduling and grant sizing methods for WDM PONs,” *IEEE/OSA Journal of Lightwave Technology*, vol. 28, no. 13, pp. 1922–1931, July 2010.
- [106] H. Song, B. W. Kim, and B. Mukherjee, “Multi-thread Polling: A Dynamic Bandwidth Distribution Scheme in Long-Reach PON,” *IEEE Journal of Selected Areas in Communications*, vol. 27, no. 2, pp. 134–142, Feb. 2009.
- [107] K. Fouli, T. Berisa, and M. Maier, “Optical Coding for Enhanced Real-Time Dynamic Bandwidth Allocation in Passive Optical Networks,” *IEEE/OSA Journal of Lightwave Technology*, vol. 27, no. 23, pp. 5376–5384, Dec. 2009.
- [108] F. Aurzada, M. Scheutzow, M. Reisslein, N. Ghazisaidi, and M. Maier, “Capacity and Delay Analysis of Next-Generation Passive Optical Networks (NG-PONs),” *IEEE Transactions on Communications*, vol. 59, no. 5, pp. 1378–1388, May 2011.
- [109] B. Kantarci and H. T. Mouftah, “Periodic GATE Optimization (PGO): A New Service Scheme for Long-Reach Passive Optical Networks,” *IEEE Systems Journal*, vol. 4, no. 2, pp. 440–448, Dec. 2010.