Department of Electrical and Computer Engineering

Multicasting and Groupcasting with Physical Layer Constraints in Metropolitan Area Networks with Mesh Topologies

Tania Panayiotou

A Dissertation

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

at the University of Cyprus

2013

APPROVAL PAGE

Tania Panayiotou

Multicasting and Groupcasting with Physical Layer Constraints in Metropolitan Area Networks with Mesh Topologies

The present Doctorate Dissertation was submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering, and was approved on November 28, 2012 by the members of the Examination Committee.

Committee Chair _____
Dr. Stavros Iezekiel

Research Supervisor _____
Dr. Georgios Ellinas

Committee Member _____
Dr. Kyriakos G. Vlachos

Committee Member _____
Dr. Christos Panayiotou

Committee Member _____
Dr. Vasos Vassiliou

Abstract

Next-generation optical networks are expected to support traffic that will be heterogeneous in nature with both unicast, as well as multicast/groupcast applications. Recent bandwidth-intensive applications that are driving the use of optical multicasting include telepresence, grid computing, telemedicine, software and video distribution for residential customers, movie broadcasts, interactive distance learning and video training, and distributed games amongst others.

In parallel to the emergence of high-bandwidth multicast applications, the trend is for next-generation mesh optical networks that are evolving from opaque to translucent, and eventually to transparent optical networks. These transparent networks are extremely desirable as they provide bit-rate, protocol, and modulation format transparency, thus providing better solutions in the implementation of the network architecture by minimizing the extra cost, power, and footprint associated with the additional transceivers present in an opaque architecture. These architectures must now also have the capability and build-in intelligence to efficiently support all types of traffic and all kinds of applications, including both unicast as well as multicast/groupcast applications that require either the entire or a fraction of the wavelength bandwidth.

In telecommunications networks, it is also essential that services are provided in an uninterrupted manner, leveraging fault recovery techniques and intelligent switching nodes to protect the network against failures. Multicast applications that carry traffic to multiple destinations are even more susceptible to failures as a single failure can result in the loss of information destined to a large number of users. Thus, providing failure recovery techniques for the multicast applications is of paramount importance in next-generation optical networks.

The objective of this dissertation is to investigate problems related to impairment-aware multicasting/groupcasting in metropolitan area mesh optical networks, when the physical layer impairments are also taken into consideration when designing and implementing the

appropriate provisioning and protection techniques. The main contributions of this dissertation are in the design and implementation of provisioning and protection techniques for the impairment-aware support of multicast and groupcast applications in transparent optical networks.

This dissertation fills an existing void in that area by formulating and developing efficient solutions for these problems in metropolitan area optical networks with mesh topologies. All techniques developed for wavelength routing, subrate grooming, and fault protection for multicast and groupcast applications in these networks were designed while taking into consideration the physical layer constraints, and by doing so they exhibited improved performance compared to all the rest of the previously developed techniques.

This dissertation presents a complete treatment to the impairment-aware multicast and groupcast provisioning and fault recovery problem by investigating the physical layer system model, node architectures and network engineering, as well as designing and evaluating a large number of heuristic algorithms that can provide these network control functions in a simple and efficient manner.

While this work focuses on metropolitan area networks, these methods and techniques can also be readily applicable to other types of networks as well, provided that the network-specific physical layer impairments are accounted for in the physical layer system model.

Περίληψη

Τα οπτικά δίκτυα επόμενης γενιάς αναμένεται ότι θα μπορούν να υποστηρίζουν ταυτόχρονα εφαρμογές που χρειάζονται μονοσημειακές αλλά και πολυσημειακές συνδέσεις. Μερικές πρόσφατες εφαρμογές που χρειάζονται μεγάλη χωρητικότητα και θα μπορούσαν να υποστηριχτούν καλύτερα από πολυσημειακές συνδέσεις είναι η ταυτόχρονη διανομή βίντεο σε πολλούς χρήστες, η εκπαίδευση εξ' αποστάσεως και η εκπαίδευση μέσω βίντεο, οι εφαρμογές τηλεπαρουσίας και τηλεϊατρικής, η αναμετάδοση ταινιών σε πολλούς χρήστες, τα διαδικτυακά παιχνίδια, καθώς και οι εφαρμογές υπολογιστικού πλέγματος.

Παράλληλα με την ανάπτυξη των πολυσημειακών εφαρμογών, τα οπτικά δίκτυα επόμενης γενιάς εξελίσσονται από μη-αμιγώς δίκτυα, σε δίκτυα με αμιγώς οπτικές περιοχές, και τελικά σε εξ' ολοκλήρου αμιγώς οπτικά δίκτυα. Τα αμιγώς οπτικά δίκτυα είναι επιθυμητά αφού μπορούν να παρέχουν διαφάνεια στους ρυθμούς μετάδοσης, στα πρωτόκολλα επικοινωνίας, και στην διαμόρφωση του σήματος. Αυτά τα δίκτυα μπορούν να προσφέρουν καλύτερες λύσεις όσο αφορά την αρχιτεκτονική των οπτικών δικτύων, αφού το κόστος υλοποίησης τους ελαχιστοποιείται, χρειάζονται λιγότερη ισχύ, καθώς και λιγότερο χώρο για την εγκατάσταση του εξοπλισμού του δικτύου, αφού πλέον δεν χρειάζονται οι επιπλέον πομποί και δέκτες που χρειάζονται για την λειτουργία ενός μη-αμιγώς οπτικού δικτύου. Επιπλέον, αυτές οι αρχιτεκτονικές πρέπει να μπορούν να υποστηρίξουν αποδοτικά εφαρμογές διάφορων τύπων συνδέσεων, συμπεριλαμβανομένων και των πολυσημεικών συνδέσεων, που είτε χρειάζονται ολόκληρη την χωρητικότητα που μπορεί να προσφέρει ένα μήκος κύματος είτε χρειάζονται μόνο ένα μέρος αυτής της χωρητικότητας.

Στα τηλεπικοινωνιακά δίκτυα, είναι επίσης σημαντική η απρόσκοπτη παροχή υπηρεσιών μέσω τεχνικών αποκατάστασης βλαβών για την προστασία του δικτύου. Οι πολυσημειακές συνδέσεις που στέλνουν πληροφορίες σε πολλούς προορισμούς ταυτόχρονα είναι περισσότερο επιρρεπείς στις βλάβες στο δίκτυο, αφού μια και μόνο

βλάβη στο δίκτυο μπορεί να οδηγήσει στην απώλεια πληροφορίας που προορίζεται για ένα μεγάλο αριθμό χρηστών. Γι' αυτό και οι τεχνικές προστασίας βλαβών που αφορούν πολυσημειακές συνδέσεις είναι εξαιρετικά σημαντικές στα οπτικά δίκτυα επόμενης γενιάς.

Στόχος αυτής της διατριβής είναι η διερεύνηση προβλημάτων που σχετίζονται με πολυσημειακές συνδέσεις σε μητροπολιτικά οπτικά δίκτυα τυχαίων τοπολογιών, όταν οι επιδράσεις του φυσικού στρώματος λαμβάνονται υπόψη κατά τον σχεδιασμό και την υλοποίηση αλγορίθμων δρομολόγησης για την εγκατάσταση και προστασία των συνδέσεων. Η κύρια συμβολή αυτής της διατριβής αφορά τον σχεδιασμό και την υλοποίηση αλγορίθμων δρομολόγησης για την εγκατάσταση και προστασία των πολυσημειακών συνδέσεων, σε αμιγώς οπτικά δίκτυα, λαμβάνοντας υπόψη τις επιδράσεις του φυσικού στρώματος.

Αυτή η διατριβή γεμίζει ένα κενό που υπάρχει στην έρευνα, αναπτύσσοντας αποδοτικές λύσεις για αυτά τα προβλήματα σε μητροπολιτικά οπτικά δίκτυα με τοπολογία πλέγματος. Όλες οι τεχνικές που αναπτύχθηκαν για την δρομολόγηση, πολυπλεξία, και προστασία των πολυσημειακών συνδέσεων, σχεδιάστηκαν λαμβάνοντας υπόψιν τους περιορισμούς του φυσικού στρώματος και παρουσιάζουν σημαντική βελτίωση στην απόδοσή τους σε σχέση με τις προηγούμενες τεχνικές που έχουν αναπτυχθεί μέχρι τώρα.

Σε αυτή την διατριβή παρουσιάζεται μια ολοκληρωμένη λύση για την εγκατάσταση και προστασία πολυσημειακών συνδέσεων σε αμιγώς οπτικά δίκτυα, μέσω της ανάπτυξης ενός μοντέλου για τον υπολογισμό των επιδράσεων του φυσικού στρώματος, της ανάπτυξης και σχεδιασμού κατάλληλων αρχιτεκτονικών κόμβων, καθώς και μέσω της ανάπτυξης και εκτίμησης ενός μεγάλου αριθμού ευριστικών αλγορίθμων που μπορούν να αυξήσουν την απόδοση αυτών των δικτύων.

Ενώ αυτή η διατριβή επικεντρώνεται σε μητροπολιτικά οπτικά δίκτυα, οι μέθοδοι και τεχνικές που αναπτύσσονται σε αυτή την διατριβή μπορούν να εφαρμοστούν και σε άλλους τύπους δικτύων, δεδομένου ότι οι επιδράσεις του φυσικού στρώματος που σχετίζονται με αυτά τα δίκτυα λαμβάνονται υπόψη στην μοντελοποίηση του φυσικού στρώματος.

Acknowledgments

First and foremost, I would like to express my deepest appreciation to my thesis advisor Prof. Georgios Ellinas for his valuable scientific advice, encouragement, and foresight. The successful completion of this thesis would have been impossible without his help, knowledge, and the many insightful discussions and suggestions. His broad research experience and enthusiasm for research was contagious and motivational for me. I could not have imagined having a better advisor and mentor for my Ph.D. study.

I am deeply grateful to Prof. Neophytos Antoniades for his generous help throughout the completion of my Ph.D. thesis. He provided valuable suggestions, comments and technical advice that made the completion of this thesis possible. It would not be an exaggeration to say that I was lucky enough to have the support and guidance from two of the best in the field of optical networking; both Profs. Georgios Ellinas and Neophytos Antoniades.

Also I would like to thank Prof. Antonis Hadjiantonis as I had the chance to collaborate with him as well; especially in the beginning of my research study he provided me with support and valuable suggestions.

Special thanks to my dissertation committee, Prof. Stavros Iezekiel, Prof. Kyriakos G. Vlachos, Prof. Christos Panayiotou, and Prof. Vasos Vassiliou for their guidance and helpful suggestions. Prof. Kyriakos G. Vlachos in particular, was the one who initially introduced me to the field of optical networks during my undergraduate thesis, something that led me to pursue this field for my graduate studies as well.

I would also like to acknowledge the Research Promotion Foundation of Cyprus, the University of Cyprus, and the KIOS Research Center for supporting my research.

I also wish to thank my family that provided me with love, patience, and support for the duration of my Ph.D work. This thesis is dedicated to my father Menelaos, my mother Lenia, my sister Chara, my grandparents Antonia and Michalakis, and my husband Charis.

Publications

Journal Publications

1. T. Panayiotou, G. Ellinas, and N. Antoniades, "Segment-Based Protection of Multicast Connections in Metropolitan Area Optical Networks with Quality-of-Transmission Considerations", *IEEE/OSA Journal of Optical Communications and Networking*, vol. 4, no. 9, pp. 692-702, September 2012.

2. G. Ellinas, N. Antoniades, T. Panayiotou, A. Hadjiantonis, and A.M. Levine, "Multicast Routing Algorithms Based on Q-Factor Physical Layer Constraints in Metro Networks", *IEEE/OSA Photonics Technology Letters*, vol. 21, no. 6, pp. 365-367, March 2009.

Conference Proceedings

1. T. Panayiotou, G. Ellinas, and N. Antoniades, "Protection Algorithms for Groupcast Sessions in Transparent Optical Networks with Mesh Topologies", *Proc. IEEE 17th International Conference on Optical Network Design and Modeling (ONDM)*, Brest, France, April 2013.

2. T. Panayiotou, G. Ellinas, and N. Antoniades, "p-Cycle-Based Protection of Multicast Connections in Metropolitan Area Optical Networks with Quality-of-Transmission Considerations", *Proc. 4th International Workshop on Reliable Networks Design and Modeling (RNDM)*, St. Petersburg, Russia, October 2012.

3. T. Panayiotou, G. Ellinas, and N. Antoniades, "Hybrid Graph-Based Traffic Grooming for Multicast Connections in Mesh Optical Networks", *Proc. IEEE International Conference on Communication Systems (ICCS)*, Singapore, November 2012.

4. T. Panayiotou, G. Ellinas, N. Antoniades, and A. Hadjiantonis, "A Novel Segment-Based Protection Algorithm for Multicast Sessions in Optical Networks with Mesh

Topologies", *Proc. IEEE/OSA Optical Fiber Communications Conference (OFC)*, Los Angeles, CA, March 2011.

5. T. Panayiotou, G. Ellinas, N. Antoniades, and A. Hadjiantonis, "Node Architecture Design and Network Engineering Impact on Optical Multicasting Based on Physical Layer Constraints", *Proc. IEEE 12th International Conference on Transparent Optical Networks (ICTON)*, Munich, Germany, June 2010.

6. T. Panayiotou, G. Ellinas, N. Antoniades, and A. M. Levine, "Designing and Engineering Metropolitan Area Transparent Optical Networks for the Provisioning of Multicast Sessions", *Proc. IEEE/OSA Optical Fiber Communications Conference (OFC)*, San Diego, CA March 2010.

7. G. Ellinas, T. Panayiotou, N. Antoniades, A. Hadjiantonis, and A.M. Levine, "Multicasting with Physical-layer Constraints in Metropolitan area Networks", *Proc. IEEE International Conference on Transparent Optical Networks (ICTON)*, Athens, Greece, June 2008.

Book Chapters

1. S. Azodolmolky, M. Angelou, I. Tomkos, T. Panayiotou, G. Ellinas, and N. Antoniades, "Impairment-Aware Optical Networking: A Survey", chapter in book titled *WDM Systems and Networks: Modeling, Simulation, Design and Engineering*, N. Antoniades, G. Ellinas, and I. Roudas (Eds), Springer Verlag, pp. 443-479, November 2011.

Contents

List of Figures

List of Tables

List of Abbreviations

ACO	Ant Colony Optimization
ADM	Add/Drop Multiplexer
ADT	Arc Disjoint Trees
AG	Auxiliary Graph
APS	Automatic Protection Switching
ASE	Amplified Spontaneous Emission
ASSP	Adaptive Shared Segment Protection
BER	Bit Error Rate
BLSR	Bidirectional Line Switched Ring
BLT	Balanced Light-Tree
BOSNR	Best-Optical Signal to Noise Ratio
CD	Chromatic Dispersion
CFT	Cycle-For-Two
CPM	Cross-Phase Modulation
DAC	Drop and Continue
D-IA-MC-RWA	Decomposed Impairment-Aware Multicast Routing and Wavelength Assignment
DMUX	Demultiplexer
DRA	Distributed Raman Amplifier
DSF	Dispersion Shifted Fiber
DWDM	Dense Wavelength-Division Multiplexing
EDFA	Erbium-Doped Fiber Amplifier
E-LAN	Ethernet LAN
ER	Efficiency Ratio
ERH	Efficiency Ratio Heuristic
FRA	Fiber Raman Amplifier

FWM	Four Wave Mixing
GA	Genetic Algorithm
GC-R	Groupcast Routing
GC-RWA	Groupcast Routing and Wavelength Assignment
GC-WA	Groupcast Wavelength Assignment
GMPLS	Generalized Multiprotocol Label Switching
GVD	Group Velocity Dispersion
HC	Hamiltonian Cycle
HDTV	High Definition Television
HG	Hybrid Graph
HST	Hybrid Steiner Tree
HT	Hybrid Tree
IA-GC-RWA	Impairment-Aware Groupcast Routing and Wavelength Assignment
IA-PGC-RWA	Impairment-Aware Protected Groupcast Routing and Wavelength Assignment
IA-HMC-RWA	Impairment-Aware Hybrid Multicast Routing and Wavelength Assignment
IA-MC-RWA	Impairment-Aware Multicast Routing and Wavelength Assignment
IA-PMC-RWA	Impairment-Aware Protected Multicast Routing and Wavelength Assignment
IA-RWA	Impairment-Aware Routing and Wavelength Assignment
ILP	Integer Linear Programming
IpC	Intelligent p-Cycle
ISI	Intersymbol Interference
ITU-T	International Telecommunication Union-Telecommunication
LDT	Link Disjoint Trees
LF	Light-Forest
LFHR	Logical First Hybrid Routing
LP	Level Protection
LTE	Line Terminating Equipment
LUF	Least-Used Light-Tree First
MCC	Minimum Cost Cycle
MCF	Minimum Free Capacity Light-Tree First

MCH	Minimum Cost Heuristic
MC-OXC	Multicast-Capable Optical Cross-Connect
MC-RWA	Multicast Routing and Wavelength Assignment
MDT	Maximum Degree Tree
MHT	Minimum Hop Tree
MILP	Mixed-Integer Linear Program
MMFA	Maximizing Minimum Freeload
MOCR	Modified Optimized Collapsed Ring
MOL	Maximum Overlapping Light-Tree
MPH	Minimum-Cost Path Heuristic
MPLS	Multiprotocol Label Switching
MCSP	Modified Conventional Segment Protection
MSSNF	Modified Segment-Based Protection With Sister Node First
MUF	Most-Used Light-Tree First
MUX	Multiplexer
MXCF	Maximum Free Capacity Light-Tree First
MXCH	Maximum Cost Heuristic Algorithm
MXOF	Maximum Overlapping Light-Tree First
NF	Noise Figure
NRZ	Non Return to Zero
OAM	Operation, Administration and Maintenance
OCR	Optimized Collapsed Ring
OEO	Optical-Electrical-Optical
OPP-SDP	Optimal Path Pair-based Shared Disjoint Paths
OSNR	Optical Signal-to-Noise Ratio
OSPT	Optimized Shortest Paths Tree
OTN	Optical Transport Network
OXC	Optical Cross-Connect
PAM	Power-Aware Multicasting
PBC	Polarization Beam Combiner
PBR	Prediction Based Routing
PBS	Polarization Beam Splitter
PCH	p-Cycles Heuristic

PDG	Polarization Dependent Gain
PDL	Polarization Dependent Loss
PFSR	Physical First Sequential Routing
PGC-RWA	Protected Groupcast Routing and Wavelength Assignment
PMC-RWA	Protected Multicast Routing and Wavelength Assignment
PLI	Physical Layer Impairment
PMD	Polarization Mode Dispersion
PP	Physical First Priority
PPH	Pruned Prim Heuristic
QBPCH	Q-based p-Cycles Heuristic
QBST	Q-based Steiner-tree
QoS	Quality-of-Service
RC	Ring Cover
RIN	Relative Intensity Noise
ROADM	Reconfigurable Optical Add-Drop Multiplexer
RX	Receiver
SBS	Stimulated Brillouin Scattering
SDS	Shared Disjoint Segments
SHR	Self-Healing Rings
SLAs	Service Level Agreements
SMF	Single Mode Fiber
SMT	Steiner Minimal Tree
SNR	Signal-to-Noise Ratio
SDH	Synchronous Digital Hierarchy
SOA	Semiconductor Optical Amplifier
SONET	Synchronous Optical Network
SP	Segment Protection
SPLB	Segment Protection with Load Balancing
SPM	Self-Phase Modulation
SPT	Shortest Path-Based Tree
SRS	Stimulated Raman Scattering
SSNF	Segment-Based Protection with Sister Node First
SST	Sub-Steiner Tree

ST	Steiner Tree
TDM	Time Division Multiplexing
TFT	Tree-for-Two
TON	Transparent Optical Network
TX	Transmitter
UPSR	Unidirectional Path-Switched Rings
VOA	Variable Optical Attenuator
VPN	Virtual Private Network
WA	Wavelength Assignment
WDM	Wavelength-Division Multiplexing
WSS	Wavelength Selective Switches

List of Notations

Be	Electrical bandwidth
Bo	Optical bandwidth
Bo_{amp}	Amplifier optical bandwidth
Bo_{filter}	Optical filter bandwidth
e	Electron charge
f	Frequency
h	Planck's constant
I_0	Current level for symbol 0
I_1	Current level for symbol 1
I_{ASE}	Current of ASE noise
NEC	Noise equivalent circuit
p_1	Eye opening-level 1
p_0	Eye opening-level 0
r	Responsivity of receiver
r_{ASE}	Optical power spectral density of ASE noise
R_b	Bit rate
RIN	Relative intensity noise
X	Extinction ratio
λ	Wavelength
σ_0	Sum of the variances for the various noise components for symbol 0
σ_1	Sum of the variances for the various noise components for symbol 1
$\sigma_{ASE-ASE}$	Variance of ASE-ASE beat noise
$\sigma_{ASE-shot}$	Variance of ASE-shot beat noise
σ_{s-ASE}	Variance of shot-ASE beat noise
σ_{shot}	Variance of shot noise
σ_{RIN}	Variance of RIN noise

σ_{th}	Variance of thermal noise

Chapter 1

Introduction

1.1 Background

The need for more and more capacity in the network to accommodate a myriad of new applications has risen dramatically over the last decade. Advances in the telecommunications industry along with the tremendous growth of the Internet and the World Wide Web have offered the end users a large amount of bandwidth-intensive applications such as e-learning, e-commerce, on-line real-time gaming, music, and video file sharing to name a few. On the other hand, the continuously increasing number of end users as well as their growing demand for information are further driving the industry to some challenging times, in order to be ready with the appropriate technologies and engineering solutions to meet the increasing bandwidth needs. This growing demand for bandwidth is expected to continue well into the foreseeable future, as it is estimated that bandwidth usage in the Internet alone is doubling every six to twelve months. Furthermore, as technological advances have succeeded in continuously reducing the cost of bandwidth, the development of new sets of applications are emerging that continuously make use of more and more bandwidth [27, 123, 149, 182].

Fiber-optic networks that are able to transmit information at virtually the speed of light, having tremendous bandwidth capabilities and low bit-error rates, are the most promising candidates to meet the explosive growth of bandwidth demand and have been hailed as the "ultimate speedway". Enormous quantities of optical fiber were deployed throughout the world in the last two decades, and fiber-optic networks are nowadays used extensively by virtually all telecommunications carriers.

Optical networks have evolved steadily over the last two decades from point-to-point systems at the physical layer providing transport capabilities through optical fibers, to ring, and subsequently mesh topologies with intelligent switching elements (reconfigurable optical add-drop multiplexers (ROADMs), optical cross-connects (OXCs), etc) that can now provide provisioning of wavelength and sub-rate connections, fault accommodation, as well as several other control functionalities at the physical (optical) layer [165].

Initially, fiber was used in point-to-point transmission links as a direct substitute for copper, with the fibers terminating on electronic equipment. These architectures, however, were limited by an electronic speed bottleneck as only a small fraction of the potentially available bandwidth could be used. Wavelength-Division Multiplexing (WDM) has emerged as the next step in the evolution of transport networks for opening up the Terahertz transmission bandwidth of single-mode optical fibers. Even though it was clear in the early years of optical fiber transmission that by multiplexing a number of wavelengths for simultaneous transmission within the same fiber the capacity of a fiber link could be increased at minimum cost, the optical switching technology necessary to convert isolated fiber transmission links to optical networks matured sufficiently in the early 2000s. Present commercial deployments include WDM optical transmission systems with 32 to 64 wavelengths per fiber with each wavelength transmitting at 10 Gbps. The transmission speed is increasingly moving to 40 and eventually to 100 Gbps, with a 50 GHz grid channel spacing [35]. Dense WDM (DWDM) transmission systems with terabits per second capacity, utilizing more than 100 wavelengths per fiber with a channel spacing of 25 GHz have been implemented today, and the number of wavelengths per fiber is expected to increase significantly in the near future [22, 123].

With the successful commercialization of WDM, the advent of DWDM, and several key technology advancements of DWDM component technologies (such as optical amplifiers, lasers, filters, and optical switches amongst others) within the optical networking space in the second half of the 1990s, the optical transport network (OTN) has been standardized by the ITU-T as the underlying infrastructure (transport layer) of the network [23]. This provides for the management of services using multiple, different wavelengths of light over the same optical fibers, with the clients of the OTN being able to occupy up to a full wavelength. The OTN provides for carrier-grade operations, administration, and maintenance (OAM) for these managed wavelength services, and protection switching for high availability. Focus on manageability has resulted in advances in control and management functionalities, including taking control techniques designed for the logical (electrical) layers of the network and adapting them to the physical (optical) layer. As an example, the control protocol known

as Generalized Multiprotocol Label Switching (GMPLS) has been proposed for application to control functions in the physical layer. GMPLS is a generalization of the Multiprotocol Label Switching (MPLS) protocol, which was designed as an improvement over the packet-forwarding techniques used in IP networks [182]. For IP networks, MPLS is used to provide the control plane necessary to ensure automated provisioning, maintain connections, and manage the network resources including providing Quality-of-Service (QoS) and Traffic Engineering (TE). With the extension of the MPLS framework to support not only devices that perform packet switching, but also those that perform switching in time, wavelength, and space, GMPLS can now be applied as the control plane for wavelength-routed optical networks [145].

Thus, the physical (optical) layer now offers and manages the capacity required to transport traffic between clients in the logical layer. It includes wavelength transmission equipment (DWDM), wavelength switching or cross-connect equipment, and wavelength grooming equipment handling subrate circuits. As an example, in Fig. 1.1, physical layer nodes are optical cross-connects (OXCs) and logical layer nodes are IP routers. A link in the logical layer is created via a lightpath between the end-nodes of that link, where a lightpath is an optical path established between a pair of source-destination nodes at the physical layer. The collection of lightpaths in the optical layer therefore defines the topology of the virtual network interconnecting the switching entities (e.g., IP/MPLS routers) in the logical layer [145]. From the logical layer perspective, a lightpath, possibly over multiple OXCs, is always viewed as a point-to-point link. For time-division-multiplexing (TDM) clients, the optical lightpath is a large structured, fixed-bandwidth pipeline.

In these types of layered architectures, two possible scenarios may exist for the control and management functionalities. The first one corresponds to the case where all of the intelligence resides within the logical layer while the other one assumes that the intelligence is shared between the logical and the optical layers. Although the first scenario may be a viable architecture, this thesis deals with the case where intelligent optical switching is present in the network. For example, utilizing the control plane at the optical layer, an optical lightpath requested by the client can now be dynamically set-up/tear-down, while the client network has no knowledge of the optical network path set-up mechanism [103]. Moving the networking functionality and intelligence down to the optical layer, is more compelling in terms of simplicity, scalability, overall cost savings, and the feasibility for near-term deployment.

Optical networks, as demonstrated in Fig. 1.2, are either opaque (with electrical components providing optical-electronic-optical (OEO) conversions at all network nodes), translu-

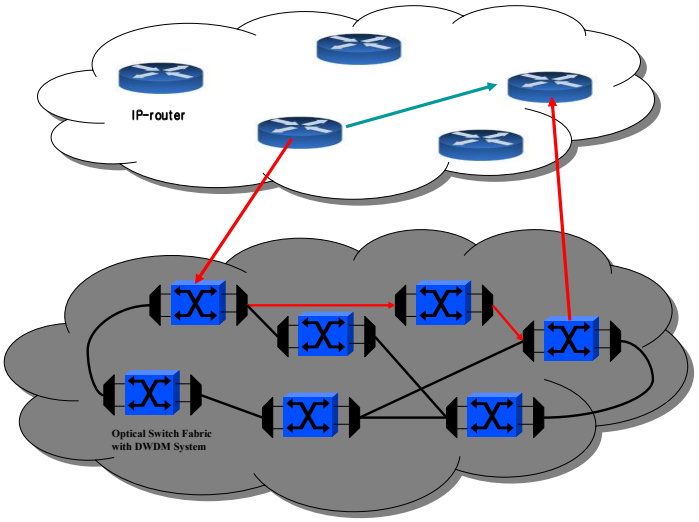

Figure 1.1: Physical (optical) and logical (IP) network layering example.

cent (where OEO conversions are provided at some of the network nodes, while some of the connections can stay in the optical domain throughout), as well as transparent (all-optical), where the nodes provide pure optical switching and the signal is never converted back to the electronic domain until it reaches the receiver at the destination node.

In the opaque approach (Fig. 1.2(a)), all switching and processing of the data at the nodes can be handled by electronics (opaque node/opaque network) or the node switch fabric can be transparent while still maintaining transponders at the WDM systems (transparent node/opaque network), thus again providing OEO conversions at all network nodes. Opaque architectures are flexible in the sense that they can provide wavelength conversion, they offer full digital regeneration of the signal, including reshaping, retiming, and resynchronization, they can provide grooming and multiplexing capabilities, they only require link-to-link engineering, they are modular and provide interoperability with legacy systems as well as between different vendors, and since they have access to the electrical signal overhead bytes they can readily provide all the necessary control and management functions. Opaque switching nodes (with an electronic switch fabric and transponders present in the WDM systems) are the ones that are currently deployed by the network operators in core optical networks. However, although this is a well-established technology, the large number of optical-to-electrical-to-optical (OEO) conversions at each switching node greatly increases the network cost, the power consumption, as well as the footprint required to deploy these switches. Furthermore, these architectures cannot keep pace with the growth in capacity

of optics in the near future and the rapidly growing customer demand for bandwidth [27]. The scaling limitations in signal bit rate, switch matrix port count, and network element cost, were the key motivations behind the attempt to develop large port-count transparent switches to be used in opaque network architectures [24]. Finally, the absence of transparency in these architectures in terms of bit rate, modulation, and protocol format makes this approach technology-dependent which means that with every technology upgrade all switching and WDM systems must be upgraded as well.

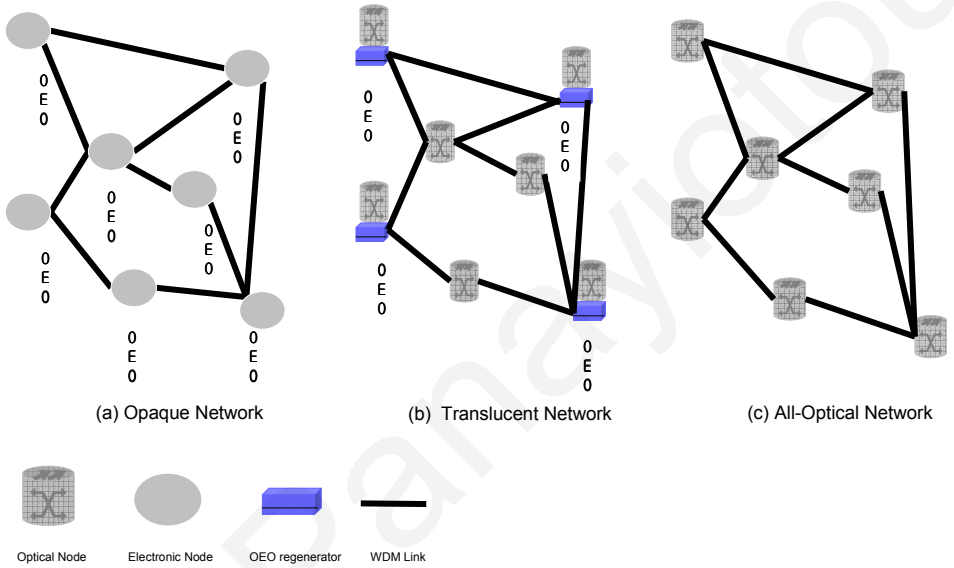

Figure 1.2: Optical network evolution [7].

Translucent network architectures that support OEO functionality at only some of the network nodes and seek a graceful balance between network design cost and service provisioning performance, have also been proposed as a solution between opaque and all-optical networks. Translucent optical networks, shown in Fig. 1.2(b), exploit the advantages of both transparent optical networks, where connections are switched in the optical domain, and opaque networks where connections are optically terminated at the intermediate nodes and switched in the electrical domain. On one hand, optical transparency offers considerable bandwidth at low cost. On the other hand, by performing opto-electronic signal regeneration at some of the intermediate nodes, it is possible to recover the signal degradation due to physical impairments and also have access to the signal overhead bytes [164]. This approach in fact eliminates much of the required electronic processing and allows a signal to remain in the optical domain for much of its path, bringing a significant cost reduction due to the

removal of electronic processing equipment [38].

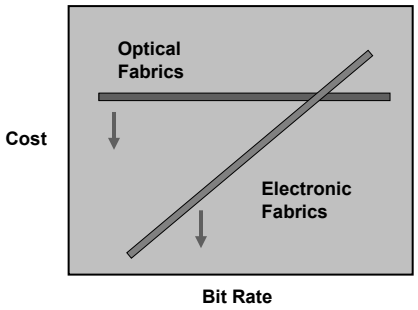

Figure 1.3: Advantages of optical switch fabrics [24].

It is envisioned that at some point in the future the network operators will eventually move to the all-optical architectures mainly driven by cost and bandwidth considerations. For example, Fig. 1.3 shows the advantages in cost that optical fabrics have at very high bit rates when compared to electronic fabrics. Even though in early stages of 2.5 Gbps and 10 Gbps development the crossover point shown in Fig. 1.3 appeared to be at the 2.5 Gbps and then the 10 Gbps rates, under today's more realistic traffic growth scenario, and given the continued decline in price of the OEO components, the crossover point has shifted to even higher bit rates [24]. However, as the traffic grows and the bit rates increase substantially, translucent (optical-bypass) core WDM networks using ROADMs and tunable lasers appear to be on the road towards widespread deployment, while all-optical networks using DWDM technology and OXCs appear to be the sole approach for transporting huge network traffic in future core (and even metro) mesh optical networks [138]. Today, optical networking over transcontinental or global distances requires electronic regeneration. However, one can still reap the many benefits of transparent optical networking by dividing a large backbone network into contained transparent domains, or islands of transparency [158].

Thus, the focus of this dissertation is on transparent (all-optical) networks with no electronic processing at any intermediate nodes as shown in Fig. 1.2(c). This network consisting of intelligent optical cross-connects is able to dynamically provision wavelength channels that are transparently routed between source and destination network nodes (*wavelength-routed networks*). Transparency in the network provides several advantages. In a transparent network, a data signal remains in the optical domain for the entire lightpath, eliminating the expensive OEO conversions. Moreover, it can offer transparency to bit rate, modulation format, as well as protocol format. Therefore, the infrastructure is future-proof in that if protocols or bit rates change, the equipment deployed in the network is still likely to be able

to support the new protocols and/or bit rates without requiring a complete overhaul of the entire network.

In transparent networks, however, there are several challenges that need to be addressed before such architectures can be deployed. Some of the main challenges for transparent optical network implementation include:

- The physical layer impairments (PLIs) incurred by the non-ideal optical transmission medium (and described in detail in Chapter 2) accumulate along an optical path limiting the transmission reach of optical signals [64, 123, 149, 165, 167]. To extend the reach of the networks, signal regeneration is essential.

- As PLIs accumulate, end-to-end system engineering is now required in the network, in contrast to their opaque counterpart, where only point-to-point system engineering is required between every node pair.

- There is limited interoperability among different vendor systems and with legacy systems that are already in the network infrastructure.

- Transparent optical networking solutions fail to recover the full functionality of the optoelectronic versions they replace without byte level access. Therefore, it is a challenge to provide the control and management functionalities (such as automated provisioning, fault recovery, multiplexing and grooming, automated network database creation, etc) that are readily available when we have access to the electrical signal. However, some of these functionalities can be addressed via clever innovation as well as standardization efforts, especially when the transparent switches are complemented by an opaque function.

Furthermore, this thesis focuses on metropolitan area optical networks. The metropolitan area network environment has lagged behind in the availability of low-cost, fast service provisioning using WDM, mainly due to the legacy infrastructure of synchronous optical network/synchronous digital hierarchy (SONET/SDH) equipment in metro regions. Most of the network functionality is provided electronically, which results in extensive optical-to-electronic and electronic-to-optical conversions at each node in the network. In contrast to long-haul optical networks, which are optimized for transmitting very high bit rates over long distances with very few add/drops, metro networks should be optimized to offer flexible connections and scalable bandwidth for new services on the optical layer. Since traffic

requirements in this environment can change constantly, it is important to have the right technology to deal with possible network reconfigurations and varying network loads. The presence of multiple customers in the metro environment having diverse requirements makes the traditional opaque network designs difficult to scale and adapt in a cost-effective way. Unlike long-haul networks, the metro networks of today are driven by central office access and transmission equipment costs, which are shared among a significantly smaller revenue base. New metro-area equipment must offer significantly increased functionality and performance at a lower cost per connection. It is only lately that the WDM technology and the optical transparency it allows have matured enough and become sufficiently cost effective to replace the traditional transponder-based designs in metro [6].

1.2 Motivation

Next-generation optical networks are expected to support traffic that will be heterogeneous in nature with both unicast and multicast/groupcast applications. For example, software and video distribution for residential customers, movie broadcasts, distributed computing, interactive distance learning and video training, video-conferencing, and distributed games are just a few applications that are today widely deployed and require point-to-multipoint connections from a source node to several destination nodes in the network. Optical multicasting has been investigated in the research community since the early days of optical networking [156, 182], but has only recently received considerable attention from the service providers, mainly because now many applications exist that can utilize the multicasting feature. Multicasting provides an easy means to deliver messages to multiple destinations without requiring too much message replication, and with a cost less than the cost of the equivalent distinct unicast connections to these destinations.

Other recent applications that will be driving the use of optical multicasting in the near future include telepresence applications that have grown considerably over the last few years in an effort to reduce travel costs and time, grid computing applications that involve coordinate resource sharing over a dynamic collection of users, as well as telemedicine applications that require the transmission of bandwidth-intensive imaging and video to multiple destinations.

Parallel to the emergence of high-bandwidth multicast applications, as discussed in the previous section, the trend is for next-generation mesh optical networks that are evolving from opaque, to translucent, and eventually to transparent optical networks (TONs) where the signal stays entirely in the optical domain from the source to the destination node, with-

out undergoing any optical-to-electrical-to-optical (OEO) conversions. These transparent networks are extremely desirable as they provide bit-rate, protocol, and modulation format transparency, thus providing better solutions in the implementation of the network architecture by minimizing the extra cost, power, and footprint associated with the additional transceivers/transponders present in an opaque architecture. These networks must now also have the capability and build-in intelligence to efficiently support all types of traffic (unicast, multicast, and groupcast) and all kinds of applications. In addition, as in today's traffic a typical multicast application requires bandwidth that could be only a fraction of the wavelength bandwidth, the capability to multiplex several independent lower-speed traffic streams (sub-wavelengths) onto a single wavelength must be provided, in order to efficiently utilize the capacity of each wavelength channel (multicast traffic grooming).

In telecommunications networks, it is also essential that services are provided in an uninterrupted manner and nowadays service level agreements (SLAs) between the network operators and the users are commonplace in trying to ensure that the customer services are protected against network failures. Especially in optical networks, which are a cable-based technology, fiber link failures (due to human error, construction work, natural catastrophes, etc) occur often and sometimes with devastating effects. Multicast applications that carry traffic to multiple destinations are even more susceptible to failures as a single failure can result in the loss of information destined to a large number of users. Thus, providing failure recovery techniques for the multicast applications is of paramount importance in next-generation optical networks. Duplicated equipment, redundant capacity in the network reserved for recovery purposes, as well as automatic protection switching protocols are generally utilized to recover from a failure condition.

Provisioning and protecting multicast connections in transparent optical networks has been extensively studied in the literature from the logical layer point of view (control plane approach) that only tries to accommodate the multicast connection without considering whether when applying the appropriate routing/grooming and/or protection techniques the solution found results in optical signals at the destination nodes that are feasible (i.e., have an acceptable signal quality). Such physical-logical interactions in transparent networks have been investigated only for unicast connections and not for multicast/groupcast connections apart from the consideration of optical power budget constraints. Furthermore, even though the last few years have marked the introduction of WDM in metro applications through work on architectural proof-of-concept, experimental demonstrations, field trials, and finally real commercial deployments [151, 166, 210], there has been no in-depth published study on de-

signing and engineering specific transparent metro network WDM architectures, capable of supporting traffic of unicast, multicast and groupcast type.

Clearly, there is a void in the research in terms of impairment-aware multicast and groupcast when the physical layer impairments are also taken into consideration when designing and implementing the appropriate provisioning and protection techniques. This void is filled by the work performed in this dissertation as explained in detail in the section that follows.

1.3 Thesis Objective/Contribution

The preceding discussion on the two new trends in next-generation optical networks, namely the drive for transparent architectures and the need to support bandwidth-intensive multicast/groupcast applications provides precisely the context for the problems and solutions developed in this dissertation.

The objective of this thesis is to study problems related to impairment-aware multicasting/groupcasting in metropolitan area mesh optical networks, when the physical layer impairments that affect the BER of a system, are also taken into consideration when designing/implementing the appropriate provisioning and protection techniques. The main contributions of this dissertation are in the design and implementation of provisioning and protection techniques for the impairment-aware support of multicast and groupcast applications in transparent optical networks.

The novelty of the work stems from the fact that while impairment-aware unicast provisioning and protection techniques have been extensively studied as amply demonstrated in Chapter 2, impairment-aware multicast provisioning and protection is a subject that has not been investigated apart from the cases where the power budget constraints are considered in the design of the techniques. The work developed in this dissertation presents the first time the physical layer impairments of the network are taken into consideration when developing the multicast and groupcast provisioning and fault recovery techniques. As all techniques developed for wavelength routing, subrate grooming, and fault protection for multicast and groupcast applications in these networks, were designed specifically to account for the additional constraints imposed by the physical layer, they exhibited improved performance compared to all the rest of the previously developed techniques that accounted only for the optical power budget or were developed irrespective of the physical layer constraints. This work shows the close interaction between the physical and logical layers needed for developing provisioning and fault recovery techniques suitable for the transparent (all-optical)

network architectures.

This dissertation presents a complete treatment to the impairment-aware multicast and groupcast provisioning and fault recovery problem by investigating the physical layer system model, designing and evaluating appropriate multicast-capable node architectures and network engineering solutions, as well as developing and evaluating a large number of heuristic algorithms that can provide these network control functions in a simple and efficient manner.

Specifically, this dissertation initially improves on the existing multicast routing and wavelength assignment (MC-RWA) heuristic algorithms by formulating novel multicast provisioning techniques that succeed in maximizing the multicast connections admitted to the network. These techniques, that now also take into consideration the physical layer impairments (PLIs), by utilizing the Q-factor as described in Chapter 2, are shown to improve the overall blocking probability compared to previous multicast routing techniques found in the literature that either do not account for the PLIs or account only for the power budget constraints. Furthermore, several multicast-capable node designs and network engineering solutions are proposed and analyzed by considering the impact of node design/network engineering on the multicast provisioning techniques developed.

As an efficient fault recovery approach is of paramount importance for the successful deployment of bandwidth-intensive multicast applications in transparent optical networks, the second significant contribution of this dissertation is the development of an innovative segment-based protection technique that was designed while accounting for the physical layer constraints. This technique outperforms all existing protection approaches in terms of blocking probability and performs even better when sharing techniques are also utilized in the network in an effort to reduce the redundant capacity requirements. This is again a clear indicator that when designing impairment-aware multicast protection techniques, the effect of PLIs cannot be ignored for solutions that require quality-of-transmission (QoT) guarantees. Additional work on fault recovery in this dissertation focused on developing cycle-based protection techniques for providing fault recovery for transparent multicast connections. Different techniques are utilized for identifying the cycles in the network graph and for finding backup paths to protect the network against failure events for networks with and without sharing capabilities. This is the first time segment- and cycle-based techniques are investigated while taking into consideration the physical-logical layer interactions, making the results extremely important, as they can provide a useful insight on whether some of the most popular protection techniques that appear in the literature are feasible when the PLIs are also taken into consideration.

A third contribution of this dissertation involves the development of traffic grooming techniques for provisioning subrate multicast connections on transparent optical networks with physical layer impairments. This is again of vital importance for the practical implementation of transparent multicast applications, as most multicast service applications require capacities that are only a fraction of the wavelength capacity (sub-wavelength connections). The grooming technique developed in this dissertation focuses on building a dynamic hybrid topology consisting of both the available physical and logical resources and then routing each arriving multicast call on the hybrid graph. Several schemes are implemented for building the hybrid graph, with each scheme prioritizing the logical resources according to a different characteristic. The hybrid routing heuristic developed, in conjunction with the hybrid graph building techniques, was shown to outperform all other existing grooming techniques that route multicast calls on logical and physical layers separately and sequentially, especially when the PLIs are also considered.

Finally, this dissertation extends the impairment-aware multicast provisioning and fault recovery techniques to account for groupcast connections as well, motivated mainly by the recent emergence of several bandwidth-intensive groupcast applications. Efficient groupcast routing and grooming heuristic algorithms were initially developed that reduce the connection cost of the "light-forests" (groupcast connectivity set) while at the same time achieving acceptable signal quality at the receivers. New groupcast protection approaches were also developed (considering the unique characteristics of the groupcast connections) that outperform all other groupcast protection approaches that are extensions of already existing multicast protections schemes. However, when the physical layer impairments are also considered, the segment-based protection technique, that was specifically designed to account for the physical layer constraints, outperforms the rest of the approaches. As this is the first time impairment-aware groupcast provisioning and protection schemes are considered in the literature, the techniques developed in this thesis serve as a reference and a first insight to subsequent researchers that will work in this area.

Clearly, the problem of impairment-aware multicast and groupcast provisioning (including wavelength routing and subrate grooming) and fault recovery is crucial in the deployment of future bandwidth-intensive applications in transparent optical networks. This dissertation is instrumental in addressing these problems by formulating and developing efficient solutions for metropolitan area optical networks with mesh topologies. Furthermore, while the discussion focuses on metropolitan area networks, these methods can also be readily applicable to other types of networks as well, provided that the network-specific physical layer

impairments are accounted for in the physical layer system model.

1.4 Organization of the Thesis

The rest of the thesis is organized as follows:

Chapter 2 is mainly focused on the impairment-aware routing and wavelength assignment (IA-RWA) problem for unicast connections. Initially, the various physical layer impairments that degrade the signal quality while traversing optical fibers and passing through network switching nodes are discussed, followed by a taxonomy and a survey of the existing IA-RWA techniques for unicast connections. The physical layer system model that is used throughout the thesis is also explained and analyzed in this chapter. This model which is based on the system Q-factor is utilized as a Quality of Transmission (QoT) measurement tool throughout the thesis.

Chapter 3 introduces the first part of the work specific to this thesis, starting with the node architecture and system engineering designs for supporting multicast calls in transparent optical networks. It then presents new impairment-aware multicast routing and wavelength assignment techniques that are designed specifically to accommodate the physical layer impairments that are present in the network and are modeled as described in Chapter 2. A number of different techniques are developed to solve the impairment-aware MC-RWA (IA-MC-RWA) problem and the impact of the node design/engineering, as well as the impact of the physical layer constraints, via the Q-factor, on the multicast routing heuristics is demonstrated. Work that specifically considers the polarization dependent gain/loss (PDG/PDL) is also considered in this chapter, followed by additional analysis and discussion on techniques that also allow for the creation of multiple sub-light-trees when provisioning a single multicast connection.

Chapter 4 is devoted on the problem of provisioning impairment-aware protected multicast connections. The focus is on fault recovery (via protection techniques) of multicast connections in WDM all-optical networks with mesh topologies. Since the predominant type of failure in these networks is a link failure (fiber cut) and the probability of more than one simultaneous failures is extremely low, all protection techniques in this chapter examine survivability upon single link failure scenarios. Segment- and cycle-based protection techniques are developed and analyzed, while also taking the physical layer impairments under consideration, and their performance is compared with a host of existing tree-based, path-based, segment-based, and cycle-based protection approaches. Self-sharing and cross-

sharing techniques are also considered in conjunction with the protection schemes, in an attempt to further reduce the redundant capacity requirements in the network.

Chapter 5 extends the work in Chapter 3 by now also considering the provisioning of subrate multicast connections as well in the transparent network environment where physical layer constraints are accounted for. Multicast-capable switching architectures supporting the grooming functionality are first presented, followed by the development of impairment-aware multicast grooming techniques that provision the multicast subrate connections on hybrid graphs comprised of both physical and logical links in the network. Several techniques are initially proposed for building the hybrid graphs, with each scheme prioritizing the logical resources according to a different characteristic. This is followed by multicast grooming/routing techniques that are used in conjunction with the hybrid graph creation approaches. Finally, performance analysis is performed to compare the newly developed impairment-aware grooming techniques to the heuristics that route the new multicast calls on logical and physical layers separately and sequentially.

Chapter 6 extends the impairment-aware multicast provisioning (routing and grooming) and protection techniques to now account for groupcast connections when the physical layer constraints are considered. Background on the groupcast routing and wavelength assignment (GC-RWA) problem is first given. This is followed by the development and performance analysis of impairment-aware GC-RWA (IA-GC-RWA) techniques. Furthermore, impairment-aware protection of groupcast sessions is examined and several groupcast protection algorithms are presented and their performance is also evaluated when the PLIs are considered. Specifically, several protection schemes initially proposed for multicast traffic, were extended to support groupcast connections, followed by the analysis of newly proposed groupcast protection schemes that were developed considering the unique characteristics of groupcast connections. Self- and cross-sharing techniques are again considered in the design of the protection techniques. Finally, in this chapter, the groupcast traffic grooming problem is investigated and a number of grooming techniques for groupcast sessions are presented and analyzed for both the case where the PLIs are not considered and the case where the PLIs are taken into account.

Chapter 7 ends the thesis by offering some concluding remarks and emphasizing the original contributions of this dissertation. Several issues that were not answered in this work are included in this chapter as topics of future work. Research directions for these challenges are also included. The list of future directions presented in this chapter is not exhaustive, but it is meant to serve as a guide towards some interesting directions that warrant investigation.

Chapter 2

Unicast Connections in Transparent Optical Networks

In a wavelength-routed WDM network, end users communicate with one another via all-optical WDM channels, which are referred to as lightpaths [215]. A lightpath is used to support a connection between a source and a destination node pair as shown in Fig. 2.1. These connections are called point-to-point or unicast connections. In order to create a connection for a unicast request, a route must be found between the source-destination pair and also a wavelength must be assigned to the route. This is the well-known routing and wavelength assignment (RWA) problem that is proven to be NP-complete [59]. Previous studies have already investigated the RWA problem as summarized in [215].

In transparent (all-optical) networks when a route between a source-destination pair is found, the wavelength assignment for that route must obey the following two rules:

Rule 1: Two connections cannot be assigned the same wavelength on the same fiber.

Rule 2: The same wavelength must be assigned along the route from the source node to the destination node in the absence of wavelength conversion in the nodes along the path.

The former constraint is known as the color-clash constraint, while the latter constraint is known as the wavelength continuity constraint. However, if wavelength conversion is present in some (or all) of the switching nodes in the network, then the wavelength continuity constraint does not hold anymore (provided that the path traverses at least one wavelength converter) and the RWA problem is reduced to just finding the minimum cost route for the unicast request (a problem that has a polynomial time solution), where the only limiting factor is the number of available wavelengths (channels) on each link. Note that throughout

the thesis it is assumed that no wavelength conversion is present, so both the color clash and wavelength continuity constraints must be met. In the network example of Fig. 2.1 it is shown that three lightpaths are established into the network while only two wavelengths are available in the network (λ_1 and λ_2). Since lightpaths ($A-D-E-F$) and ($A-D-E-H$) are sharing some common links, different wavelengths are assigned along their paths, while lightpath ($G-E-B-C$) can be established either on λ_1 or on λ_2.

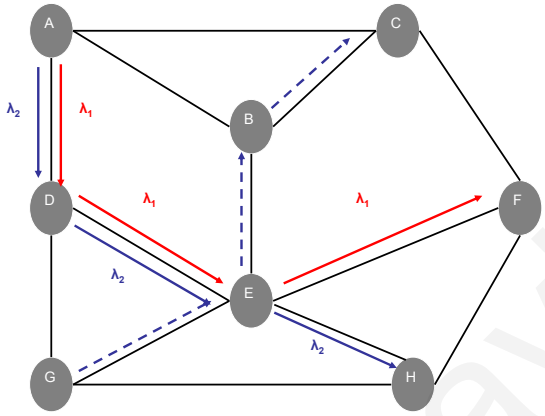

Figure 2.1: Unicast connections in a WDM optical network.

Typically, the RWA problem can be either static or dynamic. In the static case, all lightpath requests are known in advance, and the routing and wavelength assignment operations are performed off-line (during the design and planning of the network). The typical objective in this case is to maximize the number of requests accepted into the network, given a number of available wavelengths, or otherwise to minimize the total number of wavelengths required upon the establishment of every request. In its dynamic case, unicast requests arrive sequentially in the network (in real-time during the network operation), they are not known in advance, and once they are established they remain into the network for some finite amount of time and then depart. Once a connection departs, the network resources used by that connection are now released to be used by future connections. The objective of the dynamic RWA is to find routes for the unicast connection requests and assign wavelengths to these routes in such a way that the call blocking probability is minimized. In both dynamic and static cases, requests are blocked if there is no available wavelength in the network to serve the request without violating the aforementioned two rules.

For the static RWA problem, integer linear programming (ILP) formulations can be used to find optimal solutions [215]. For example, an ILP formulation is shown below, where

the objective function is to maximize the number of established connections given a fixed number of wavelengths and a given set of unicast connection requests. For this formulation the following definitions are used:

- N_{sd}: Number of source-destination pairs.
- L: Number of links.
- W: Number of wavelengths per link.
- $m = \{m_i\}$, $i = 1, 2, ..., N_{sd}$: Number of connections established for source-destination pair i.
- ρ: Offered load (total number of connection requests to be routed).
- $q = \{q_i\}$, $i = 1, 2, ..., N_{sd}$: Fraction of the load which arrives for source-destination pair i.
- $q_i\rho$: number of connections to be set up for source-destination pair i.
- P: Set of paths on which a connection can be routed.
- $A = (a_{ij})$: $P \times N_{sd}$ matrix in which $a_{ij} = 1$ if path i is between source-destination pair j, and $a_i = 0$ otherwise.
- $B = (b_{ij})$: $P \times L$ matrix in which $b_{ij} = 1$ if link j is on path i, and $b_{ij} = 0$ otherwise.
- $X = (x_{ij})$: $P \times W$ route and wavelength assignment matrix, such that $x_{ij} = 1$ if wavelength j is assigned to path i, and $x_{ij} = 0$ otherwise.

The objective of the routing and wavelength assignment problem is to maximize the number of established connections, $X_0(\rho, q)$. The ILP formulation is as follows:

$$\textbf{Maximize} \qquad X_0(\rho, q) = \sum_{i=1}^{N_{sd}} m_i \tag{2.1a}$$

$$\textit{subject to}: \qquad m_i \geq 0, \qquad \textit{integer}, i = 1, 2, ..., N_{sd} \tag{2.1b}$$

$$x_{ij} \in \{0, 1\} \qquad i = 1, 2, ..., P, j = 1, 2, ..., W \tag{2.1c}$$

$$X^T B \leq 1_{W \times L} \tag{2.1d}$$

$$m \leq 1_W X^T A \tag{2.1e}$$

$$m_i \leq q_i \rho, \qquad i = 1, 2, ..., N_{sd} \tag{2.1f}$$

In this formulation Eq. (2.1a) gives the total number of established connections in the network, Eq. (2.1d) specifies that a wavelength can be used at most once on a given link ($1_{W\times L}$ is the $W \times L$ matrix whose elements are unity), and Eqs. (2.1e) and (2.1f) ensure that the number of established connections is less than the number of requested connections (where 1_W is now the $1 \times W$ matrix whose elements are unity).

To make the problem tractable, the RWA problem can be decomposed into two sub-problems, namely the routing (R) and wavelength assignment (WA) sub-problems and heuristic algorithms can be designed to solve each problem separately [150, 225]. For example, authors in [225] proposed a greedy allocation algorithm that iteratively attempts to assign each wavelength to as many connections as possible such that no two connections use a common wavelength. Specifically, the objective of the heuristic is to maximize the sum of the light-paths in the network, by first assigning a wavelength to the optical connection that requires the greatest number of links between its source-destination pair, followed by the connection that requires the next greatest number of links among the connections which do not use the links used by the first connection, and so on, until no more connections can be assigned. In [150], another heuristic approach is proposed that once it calculates the shortest-paths between each source-destination pair it orders them in some manner. The set of wavelengths is also ordered in some manner and a new connection is routed on the first path on which a wavelength is available. Among the set of available wavelengths on that path the first one is selected. If no path can be found, the connection is considered blocked. Heuristic algorithms proposed for the static RWA problem however, need to be executed only infrequently, i.e., when the long-term average traffic pattern has changed, and the basic optical network architecture can therefore be scaled to very large configurations. For traffic patterns that frequently change over time, the trade-off between complexity and efficiency of the algorithms must be considered, since connections must be reconfigured according to the heuristics upon every change of the traffic pattern.

For the dynamic RWA problem, heuristics methods are generally employed for both the routing and the wavelength assignment sub-problem. For the wavelength assignment sub-problem a number of heuristics, such as Random, First-Fit, Least-Used, Most-Used, Least Loaded, MAX-SUM, etc., that require centralized network control have been proposed in [15, 21, 39, 80, 85, 183, 223]. The most common wavelength assignment heuristics that attempt to reduce the overall blocking probability for new connections are briefly described below:

- *Random*: In this scheme, among the available wavelengths one is chosen randomly, usually with uniform probability.

- *First-Fit*: In this scheme, all wavelengths are numbered. When searching for available wavelengths, a lower numbered wavelength is considered before a higher numbered wavelength. The first available wavelength is then selected. This technique is preferred in practice as it performs well in terms of blocking probability and fairness while requiring small computational overhead and exhibiting low complexity.

- *Least-Used*: This scheme selects the wavelength that is the least-used in the network, thereby attempting to balance the load amongst all wavelengths. The performance of the Least-Used approach is worse than the other three techniques described here, while also requiring additional storage and computational cost.

- *Most-Used*: The Most-Used technique attempts to select the most-used wavelength in the network (max wavelength reuse). This technique slightly outperforms First-Fit and significantly outperforms the Least-Used approach. However, it also requires additional storage and computational cost when compared to the First-Fit approach.

Throughout the thesis, the *First-Fit* wavelength assignment scheme will be considered due to its small computational overhead and low complexity and due to the fact that it also performs well in terms of blocking probability (similar results to the Most-Used technique which is the best wavelength assignment technique amongst them).

For the routing sub-problem, several algorithms can be found in the literature such as Dijkstra's [51] and Bellman-Ford's [14] shortest path algorithms, that calculate the shortest (min-cost) path for each source-destination pair in polynomial time. In general, for the calculation of the shortest-path, weights that are assigned to each network link are used by the heuristic algorithms for the computation of the path that has the minimum overall cost among every possible path for a specific source-destination pair. Link weights can represent a variety of parameters, such as the physical distance, physical impairment-related information, latency, link congestion, etc, depending on the parameter in the network that we are trying to optimize.

Dynamic RWA algorithms can be based on fixed-routing or fixed-alternate routing heuristics. Dynamic fixed-routing techniques operate on the precomputed shortest-paths for each possible source-destination pair in the network. Upon an arrival request a wavelength assignment algorithm is used to search for a valid wavelength that can serve the request. On

the other hand, dynamic fixed-alternate routing schemes operate on a predetermined number of precomputed alternative routes for each possible source-destination pair. These routes may include the shortest-path route, the second-shortest-path route, the third-shortest-path route, etc. Upon an arrival request, alternative routes are searched sequentially for a valid wavelength assignment [215].

Another approach that exists for the dynamic RWA problem, is to solve the routing and wavelength assignment sub-problems jointly. In this approach the routes can be decided according to the network state at the time of the newly arrived connection, resulting in lower call blocking than fixed and fixed-alternate routing. One possible way to solve the dynamic RWA jointly, is to consider that each wavelength in the network corresponds to a copy of the network, consisting initially of the same physical topology. Each time a request is established into a certain wavelength in the network, the link weights occupied by the request are updated to an infinite value at the corresponding replica of the network, to declare the links as unavailable. In the same way, when the request is released from a certain wavelength in the network, the link weights which were occupied by the request are updated to their initial value at the corresponding replica of the network, to declare the links as available. Therefore, upon the arrival of call, the request is routed according to the current state of the network. The wavelengths and consequently their corresponding replicas of the graph can be sorted in a list of wavelengths/replicas based on one of the wavelength assignment algorithms mentioned above, such as First-Fit, Most-Used, etc. The list of wavelengths is updated each time a call is accepted/released to/from the network.

The classical RWA problem briefly described above assumes a transparent network in which the transmission medium is perfect, and therefore every request for which a path and a wavelength assignment exists, is considered valid and feasible. This has been the state-of-the-art for many years [215], as there was no attempt to integrate the physical and logical layers when solving networking problems. In a real system, however, physical layer impairments (PLIs) degrade the signal as it is transported from source to destination, and this degradation in the worst-case scenario could be so severe that the signal cannot be detected at the receiver [134, 144, 154]. For this reason, PLIs must be taken into account during the provisioning phase of the requests. Recent work on the routing and wavelength assignment of unicast requests in transparent optical networks, has taken into account the PLIs. One way to take into account these constraints, is to consider the PLIs as constraints for the RWA decisions. In this case, the routing and wavelength assignment problem can be solved as described above, with the constraint that if the PLIs on a lightpath result in the degradation

of the signal quality at the receiver to an unacceptable level, the lightpath is not accepted into the network (it is blocked). Another way is to consider these impairments during the RWA decisions. In this case, the routing and wavelength assignment algorithms aim at minimizing the impact of the physical layer effects and they can also permit the calculation of alternate routes when considering the effect of the impairments.

The next section describes some of the most important PLIs that will be considered when devising appropriate RWA techniques that also account for the physical layer impairments.

2.1 Physical Layer Impairments

Bit Error Rate (BER) is usually used as the cost metric for the evaluation of the signal quality of a lightpath, as it takes all impairment effects into consideration. Since the BER is not readily available before the lightpath is actually set up, the PLIs are modeled instead, to statistically evaluate the BER in advance. The Q-factor is usually used as a QoT metric to model the PLIs for a target BER. The PLIs can be classified into linear and nonlinear impairments. Even though some physical layer impairments such as fiber attenuation, amplified spontaneous emission noise, chromatic dispersion, polarization mode dispersion etc., can be known ahead of time by the operator according to realistic specifications of the components in the network, some other PLIs, such as nonlinear effects cannot be calculated upfront, and thus one way of considering them during the engineering of a network is to use worst-case budget values that were previously estimated on real networks. Work in [165] presents a detailed overview of various PLIs. The most important of them are described below:

2.1.1 Optical Fiber Attenuation

The ideal optical fiber is considered as a transmission medium without losses. In real optical fibers, however, the optical signal encounters attenuation losses as it traverses through the optical medium. Attenuation losses can be modeled as shown in Eq. (2.2), where P_{out} represents the output power at the end of a fiber of length L, P_{in} represents the input power, and parameter a represents the fiber attenuation which is usually expressed in units of dB/Km. In modern fibers the minimum attenuation of around 0.2 dB/Km, is achieved in the 1550 nm range.

$$P_{out} = P_{in} \times e^{-aL} \tag{2.2}$$

The two main loss mechanisms in an optical fiber are *material absorption* and *Rayleigh scattering*. Material absorption includes absorption by silica as well as impurities in the fiber [149]. In today's fibers the material absorption of pure silica is negligible in the entire $0.8-1.6\ \mu m$ band that is used for optical communication systems. Therefore, Rayleigh scattering is the dominant loss mechanism. Rayleigh scattering arises because of fluctuations in the density of the medium at the microscopic level. The loss coefficient α_r due to Rayleigh scattering at a wavelength λ can be written as shown in Eq. (2.3). As Eq. (2.3) implies, Rayleigh scattering loss decreases rapidly with increasing wavelength due to the λ^{-4} dependence. Therefore, Rayleigh scattering is the dominant loss factor at short wavelengths. This factor, along with the significant material absorption at wavelengths longer than $1.55\ \mu m$, limit the usable optical bandwidth spectrum in the range of approximately $0.8-1.7\ \mu m$.

$$\alpha_r = \frac{A}{\lambda^4} \tag{2.3}$$

2.1.2 Optical fiber Dispersion

Dispersion is the widening of a pulse duration as it travels through a fiber. This effect arises when different components of the transmitted signal travel at different velocities in the fiber, arriving at different times at the receiver. Pulse widening can lead to intersymbol interference (ISI) if the pulse broadens enough to interfere with neighboring pulses (bits) on the fiber. ISI increases significantly the bit error rate (BER) and therefore dispersion sets a limit on the bit rate and the maximum transmission rate on a fiber-optic channel [123]. Three important forms of dispersion are described below:

- *Intermodal Dispersion*: This form of dispersion is caused as multiple modes of the same signal propagate using different paths along the fiber. Single mode fibers eliminate intermodal dispersion and can support transmission over much longer distances compared to multimode fibers where this form of dispersion is introduced. In multimode fiber, each mode propagates on a different path due to different angles of incidence at the core-cladding boundary. This effect causes different rays of light from the same source to arrive at the other end of the fiber at different times, resulting in pulse spreading. Intermodal dispersion increases with the propagation distance. Graded-index multimode fibers can be used to reduce the effect of intermodal dispersion, as in these fibers the region between the cladding and the core of the fiber consists of a series of gradual changes in the index of the refraction.

- *Chromatic Dispersion (CD)*: This form of dispersion is caused when different wavelengths travel at different speeds, even within the same mode. Thus, if the transmitted signal consists of more than one wavelength, certain wavelengths will propagate faster than others. Chromatic dispersion is caused because the refraction index is a function of the wavelength and no laser source can transmit a signal that consists of just a single wavelength. Forms of chromatic dispersion include *material dispersion* and *waveguide dispersion*. Material dispersion is caused due to the different wavelength speeds in a material. Waveguide dispersion results due to the waveguide characteristics, such as the indices and shape of the fiber core and cladding, that affect the propagation speed of the different wavelengths.

  Note that CD is a collective effect of material and waveguide dispersion [7]. Depending on the manufacturing process and the radial structure of the fiber, different types of fibers with various group velocity dispersion (GVD) can be designed. In standard single-mode fibers (SSMF) the GVD is zero for 1.3 μm. The fiber design also allows shifting the point of zero dispersion resulting in dispersion shifted fibers (DSF) that exhibit zero-dispersion at 1.5 μm.

- *Polarization Mode Dispersion (PMD)*: Single-mode fibers support two perpendicular polarizations of the original transmitted signal. Due to imperfections of commercial fibers, the two perpendicular polarizations may travel at different speeds resulting in pulse widening at the end of the fiber, which limits the bandwidth of the transmission system. The main difference between PMD and the other fiber propagating effects is that PMD shows a strong statistical behavior that is frequency dependent, changes randomly with time, and from fiber to fiber [46]. Hence, it is very time-consuming to measure the influence of PMD on implemented systems and design countermeasures. Furthermore, even though single-mode fibers can perfectly eliminate several types of dispersion, compared to the multimode fibers, PMD and CD are the two forms of dispersion that single-mode fibers, cannot as of yet eliminate.

2.1.3 Polarization Dependent Loss/Gain (PDL/PDG)

Apart from the dispersion effect, that the different states of polarization introduce in an optical fiber, another two important effects arise in passive optical components due to polarization. These effects are called *Polarization Dependent Loss* and *Polarization Dependent Gain* (PDL/PDG). PDL/PDG (measured in dB) describe the difference between the maxi-

mum and minimum loss/gain with respect to all possible states of polarization. Specifically, PDL refers to the loss of passive optical components, such as optical couplers, isolators, filters, etc., that varies as the polarization state of the propagating wavelength changes. This loss is wavelength dependent. More precisely, as an optical signal passes through an optical component, it suffers power reduction in selective directions due to spatial polarization interaction. The polarization state of the light is in general affected to some degree by all optical components. PDL is expressed as the difference between the maximum and minimum loss in dB:

$$PDL_{dB} = 10log_{10}\frac{P_{max}}{P_{min}} \tag{2.4}$$

with P_{max} and P_{min} being the measured output powers. Note that those two states of polarization always represent orthogonal polarizations [50]. At low data rates, PDL is a minor contributor to loss. However, at high bit rates (10 Gbit/s and above), PDL becomes comparable to insertion loss. Therefore, at high bit rates PDL needs careful examination. PDG refers to the dependence of the gain on the polarization of the signal and its definition is similar to that of PDL in Eq. (2.4). This sensitivity is expressed as the gain difference between the states of polarization in dB and it usually appears in optical amplifiers such as Erbium-doped fiber amplifiers (EDFAs), semiconductor optical amplifiers (SOAs), and fiber Raman amplifiers (FRAs) or distributed Raman amplifiers (DRAs).

In general, PDL/PDG affect the signal quality and system performance, since a polarized signal that passes through an optical component encounters different loss/gain compared to the unpolarized noise. Therefore, the signal to noise ratio (SNR) is affected. Furthermore, PMD along with PDL/PDG effects can produce significant amounts of signal distortions, and thus, limit the system reach. However, the PDL/PDG of concatenated components cannot be determined by just adding the PDL/PDG of the individual components, as the state of polarization of each PDL/PDG element is different due to the random birefringence of fiber links connecting these individual components. A statistical model approach is more appropriate in order to estimate the average PDL/PDG value and its variation for a given link [7].

2.1.4 Amplified Spontaneous Emission Noise (ASE)

Several technologies for optical amplification have been introduced over the years, such as Erbium-doped fiber amplifiers (EDFAs), semiconductor optical amplifiers (SOAs), and fiber Raman amplifiers (FRAs), or distributed Raman amplifiers (DRAs). The invention of

EDFAs, along with the low-loss optical fiber and the semiconductor laser, are mainly credited for the rapid development of fiber-optic communication systems. In general, EDFAs that operate over a wide wavelength range, provide flat gain to each wavelength channel and negligible crosstalk between channels, and can be used as in-line and pre-node amplifiers (to compensate for the fiber loses), as post-node amplifiers (to compensate for the node insertion losses) and as pre-amplifiers at the receiver end. A detailed description on the existing EDFA technologies can be found in [7].

Apart from amplifying the optical signals, EDFAs also introduce noise (amplified spontaneous emission (ASE) noise) due to the spontaneous transitions of photons from the upper energy levels to the ground energy levels along the doped fiber. ASE noise is unpolarized and builds up in the forward and backward directions. The amount of ASE noise created at each end of the doped fiber depends on the local population inversion and is the primary source of additive noise in optically amplified systems. ASE noise is usually quantified with the noise figure (NF) parameter (often specified in decibels (dB)) that is defined as the degradation of the electrical signal-to-noise ratio (SNR) due to the optical amplifier when measured with an ideal photodetector (Eq. (2.5)).

$$NF = \frac{SNR_{in}}{SNR_{out}} \tag{2.5}$$

In general, the ASE effect can lead to degradation of the amplifier's performance by reducing the gain of the amplifier. This effect sets a limitation on the achievable gain of the amplifier and increases its noise level. Furthermore, excess ASE is an unwanted effect in lasers, since it dissipates some of the laser's power. The ASE noise mixes with the optical signal and produces beat noise components at the square-law receiver. The ASE noise covers a large spectrum (around 40 nm) and needs to be carefully analyzed to evaluate its degrading effect on the system performance. ASE effects may be mitigated by increasing the input laser intensity or decreasing the amplifier facet reflectivities amongst other techniques. Note that one important performance parameter is the optical signal-to-noise ratio (OSNR) which is defined as the ratio of optical signal power and optical noise power as shown in Eq. (2.6),

$$OSNR = \frac{P_{out}}{P_{ASE}} \tag{2.6}$$

where P_{out} is the optical signal power and P_{ASE} is the ASE noise power measured over a specific bandwidth.

2.1.5 Noise in Photodetectors

Apart from the ASE noise, several other source of noises exist that affect a photodetector. The most important of them are shortly described below.

- *Shot Noise:* Light is made of particles (photons), which are emitted by the source at random. For that reason, the amount of photons emitted by the source (e.g., laser) is not constant, but exhibits detectable statistical fluctuations. This statistical fluctuation is the cause of the shot noise. Because of its nature, it does not depends on the quality of the detector and is unavoidable. However, the shot noise becomes a real issue only when the optical intensity is fairly low: in this case quantum fluctuations become much more noticeable. The random process of light emission can generally be modeled using a Poisson distribution. Since shot noise is a Poisson process due to the finite charge of an electron, one can compute the root mean square current fluctuations as,

$$\sigma_{shot} = \sqrt{2 \times e \times I \times B_e} \tag{2.7}$$

  where e is the electron charge, B_e is the bandwidth in hertz over which the noise is considered, and I is the DC current flowing.

- *Thermal Noise:* Thermal noise, also called Johnson noise or Nyquist noise, is the electronic noise generated by the thermal agitation of the electrons inside an electrical conductor at equilibrium, which happens regardless of any applied voltage. A device (a photodiode for instance) thermal noise can be modeled as a voltage source $V_{th}(t)$ in series with an ideal resistor R. $V_{th}(t)$ has a Gaussian distribution with a mean value of zero. It can also be modeled as current source in parallel with R and has a root mean square,

$$\sigma_{th} = \sqrt{\frac{4k_B T}{R} B_e} \tag{2.8}$$

  where k_B is the Boltzmann's constant in joules per kelvin, T is the resistor's absolute temperature in kelvins, and B_e is the bandwidth in hertz over which the noise is considered.

2.1.6 Nonlinear Effects

Although nonlinear effects in optical fiber systems are not important when operating at bit rates up to about 2.5 Gb/s, at high bit rates such as 10 Gb/s and above, nonlinear effects become important and it is essential to take them into consideration. In the case of WDM systems, however, nonlinear effects can become important even when operating at low bit rates and power [149]. Nonlinear effects in a WDM system place limitations on the spacing between adjacent wavelengths, the maximum bit rate, the system reach, and they limit the maximum power per channel [123]. Some of the nonlinear effects that are important in WDM systems are described below.

- *Stimulated Raman Scattering*: Stimulated Raman Scattering (SRS) causes power to be transferred from the lower wavelength channels to the higher wavelength channels when two or more signals at different wavelengths are injected into a fiber. This transfer of energy, corresponds to emission of photons of lower energy caused by photons of higher energy. In multiwavelength systems, the lower wavelength channels lose some power to each of the higher wavelength channels within the Raman gain spectrum, which covers a range of about 40 THz below the frequency of the input light. To reduce the amount of loss, the power on each channel needs to be below a certain level. Although SRS between channels in a WDM system is harmful to the system, SRS can also be used to provide amplification in the system, which benefits the overall system performance.

- *Stimulated Brillouin Scattering* : In the case of Stimulated Brillouin Scattering (SBS) the power lost in the scattering process is transferred to an acoustic wave. In this case the scattered wave and the acoustic wave both propagate in the opposite direction of the input light [181]. Unlike SRS, SBS does not cause any interaction between different wavelengths, as long as the wavelength spacing is much greater than 20 MHz, but can create significant distortion within a single channel. SBS produces gain in the opposite direction of the input light, back towards the source. Thus, the transmitted signal weakens, and the signal back towards the transmitter must be shielded by an isolator. To counter the effects of SBS, the input power must be below a certain threshold. In multiwavelength systems, SBS may also induce crosstalk, which can be easily avoided due to the narrow gain bandwidth of SBS.

- *Four Wave Mixing*: In a WDM system using the angular frequencies $\omega_1, .., \omega_n$, the intensity dependence of the refractive index gives rise to signals at new frequencies such as $\omega_4 = \omega_1 + \omega_2 - \omega_3$. Four Wave Mixing (FWM), causes inter-channel crosstalk and is critically dependent on the channel spacing and fiber chromatic dispersion. Decreasing the channel spacing or the chromatic dispersion, causes the FWM effect to increase. Thus, the effects of FWM must be considered even for low bit rates when the channels are closely spaced and/or dispersion-shifted fibers are used.

- *Self-Phase Modulation/Cross-Phase Modulation*: Self-Phase Modulation (SPM) arises because the refractive index of the fiber has an intensity-dependent component. This nonlinear refractive index causes an induced phase shift that is proportional to the intensity of the pulse. Therefore, different parts of the pulse undergo different phase shifts, which gives rise to chirping of the pulses. Pulse chirping in turn enhances the pulse-broadening effect of chromatic dispersion [149]. SPM effects are more significant in systems using high transmitted power and/or high bit rates and therefore must be considered. When more than one signal is present, the nonlinear interactions between the signals can produce a related phenomenon, referred to as Cross-Phase Modulation (CPM). CPM depends on the aggregate power in all signals and becomes more severe as more signals are superimposed on each other in WDM systems [181]. The effects of CPM, however, are negligible in standard single-mode fiber operating in the 1550 nm band with 100 GHz channel spacing.

2.1.7 Transmitter-Induced Signal Degradation Effects

- *Laser Noise:* Due to spontaneous emission taking place inside semiconductor lasers used for direct modulation or as optical light sources for external modulators, photons with random phase are added to the coherent output field which is generated by stimulated emission inside the laser, creating small perturbations of the amplitude and phase. Random variations of the laser intensity limit the maximum achievable OSNR while random variations of the phase lead to an increased spectral linewidth of the laser.

  The relative intensity noise (RIN) describes the ratio of the mean-squared fluctuations in the observed power spectral density and the average output power of the laser. The amount of RIN is independent of attenuation along a link, as both the nominal signal power and the noise are attenuated equally. Relative intensity noise is measured by

sampling the output current of a photodetector over time and transforming this data set into frequency with a fast Fourier transform. Alternatively, it can be measured by analyzing the spectrum of the photodetected signal using an electrical spectrum analyzer. Noise observed in the electrical domain is proportional to electrical current squared and hence to optical power squared. Therefore, RIN is usually presented as relative fluctuation in the square of the optical power in decibels per hertz over the RIN bandwidth and at one or several optical intensities. It may also be specified as a percentage, a value that represents the relative fluctuations per Hz multiplied by the RIN bandwidth. RIN represents a fundamental limit to the transmission capacity of high-speed links and the carrier-to-noise performance of analog links [7].

- *Chirp-Induced Penalty:* On SMF, the chirp produced by direct modulation of a laser interacts with fiber dispersion to distort and spread out the data pulses, causing ISI and increased BER. The chirp-induced penalty can be estimated qualitatively by evaluating the error probability for different receiver types. The chirp/dispersion interaction was analyzed using a small signal model [91]. However, this does not appear to be applicable to the large excursions in a data pulse. Instead, a phenomenological model was developed, based on experimental measurements of the peak-to-peak chirp and the chirp duration parameters [6]. This simple model describes the chirp/dispersion interactions and their effect on the level of the received 1s and 0s and provides accurate results in simulations.

2.1.8 Crosstalk

An important source of performance degradation is the interference from unwanted signals, referred to as crosstalk. Crosstalk is a phenomenon in communication systems by which power from one channel "leaks" to another channel. The "leakage" power is an unwanted term which degrades signal and system quality. Three types of crosstalk exist, namely inter-channel crosstalk, co-channel heterodyne crosstalk, and co-channel homodyne crosstalk. The different forms of crosstalk depend on the value of $| \ v_s - v_x \ |$, where v_s and v_x represent the signal and interferer optical frequencies, respectively. Inter-channel crosstalk is caused when the nominal optical frequencies are different. Co-channel heterodyne crosstalk appears when the nominal frequencies are the same but the absolute frequencies are slightly different and usually occurs at the switch node. Co-channel homodyne crosstalk is caused when both frequencies are exactly the same. The latter form of crosstalk occurs only if the

two signals come from the same source.

Co-channel crosstalk is regarded as the most difficult to alleviate and it is rendered as the most severe degradation crosstalk factor in MANs. Co-channel crosstalk has to do with the amount of "leakage" of optical power from one data stream to another on the same wavelength channel. Fig. 2.2 exhibits this scenario. The dashed black and the solid black data streams are both on the same wavelength channel λ_1, and they are arriving from diverse interfaces. At the λ_1 switch, however, some of the optical power from lightpath 1 "leaks"' to lightpath 2 and vice versa.

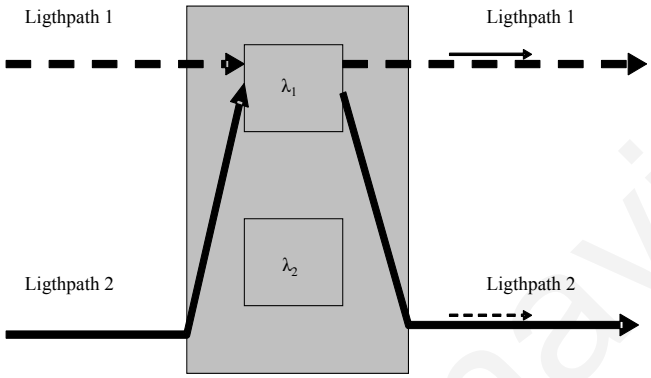

Figure 2.2: Example of co-channel heterodyne crosstalk occurring at a switch.

Similar to any other type of signal impairment, a convenient measure of crosstalk is the impairment-induced power penalty which indicates how much additional signal power is required to maintain a specified BER in the presence of the particular impairment [182]. Eq.(2.9) describes the impairment-induced power penalty, expressed in decibels.

$$P = 10\log_{10}\left[\frac{Power\ required\ with\ impairment}{Power\ required\ without\ impairment}\right] \tag{2.9}$$

However, the precise relation between crosstalk and BER depends on many factors, such as modulation format, phase, and polarization of optical signals. Procedures have been developed to take into account these factors to evaluate crosstalk-induced penalties semi-analytically [84].

2.1.9 Signal Power Divergence

Although it is desirable for the signal power or OSNR in each wavelength channel to be equal in a WDM system, non-ideal and uneven loss/gain functions of optical components

result to a signal power or OSNR imbalance. Such optical components can be optical amplifiers, filters, MUX/DMUXs, and couplers, while uneven loss/gain can be introduced from PDG/PDL effects described above, and/or from dynamic reconfigurations of the network that adds/drops channels at different power levels. Signal power divergence accumulates as the channels propagate through an optical path, affecting significantly the network performance. Adverse effects are caused in the low-power as well as the high-power channels. The low-power channels are affected the most by receiver electrical noise and interference within the network, causing low OSNR. The high-power channels can potentially suffer from fiber non-linear effects. The dynamic range limits of the optical receivers can also affect performance. The signal power divergence becomes worse through long amplifier chains as the strongest growing channels compete for gain with the weakest channels, and the unpolarized noise is attenuated differently than the signals [182].

2.1.10 Optical Filter Concatenation: Distortion-Induced Penalty

The penalty due to WDM filter concatenation is introduced as the optical signal passes through a number of filters. In transparent optical networks this effect becomes important as a signal may be demultiplexed and remultiplexed at many network elements along its path before it reaches the receiver. Optical MUX/DMUX concatenations result in signal attenuation and distortion leading to ISI. These effects set a limit on the maximum number of optical network elements that can be cascaded. Work in [152] proposed a comprehensive physical layer model including the effects of optical filter concatenation. This model is based on a semi-analytic technique for the evaluation of the error probability at the receiver. The error probability evaluation takes into account arbitrary pulse shapes, arbitrary optical MUX/DMUX and electronic low-pass filter transfer functions, and non-Gaussian photocurrent statistics at the output of a pre-amplified direct-detection receiver.

2.2 Impairment-Aware Routing and Wavelength Assignment Algorithms

As previously discussed, the classical RWA problem is NP-complete. The impairment-aware routing and wavelength assignment (IA-RWA) problem is also NP-complete and it also introduces additional complexity to the standard RWA problem since it involves a number of physical layer-related constraints that must be taken into consideration. As the PLIs are usu-

ally verified using complex analytical models of physical impairments, the complexity of the proposed IA-RWA algorithms must be a parameter of extreme importance. Most of the IA-RWA algorithms reported in the literature are based on simple heuristics. Similar to the RWA problem, the IA-RWA problem can be decomposed into the routing and wavelength assignment sub-problems, for both the dynamic and static cases, or it can be solved jointly in the dynamic case as previously described. In general, three main approaches have been considered in the literature and can be applied for both static and dynamic cases for the impairment-aware RWA as detailed below.

Case A: Compute the route and the wavelength without taking into account the PLIs and then verify the QoT of the selected lightpath considering the PLIs.

Case B: Take into account the PLIs in the routing and/or wavelength assignment decisions.

Case C: Take into account the PLIs in the routing and/or wavelength assignment decisions and also verify the QoT of the candidate lightpath before establishing the lightpath into the network.

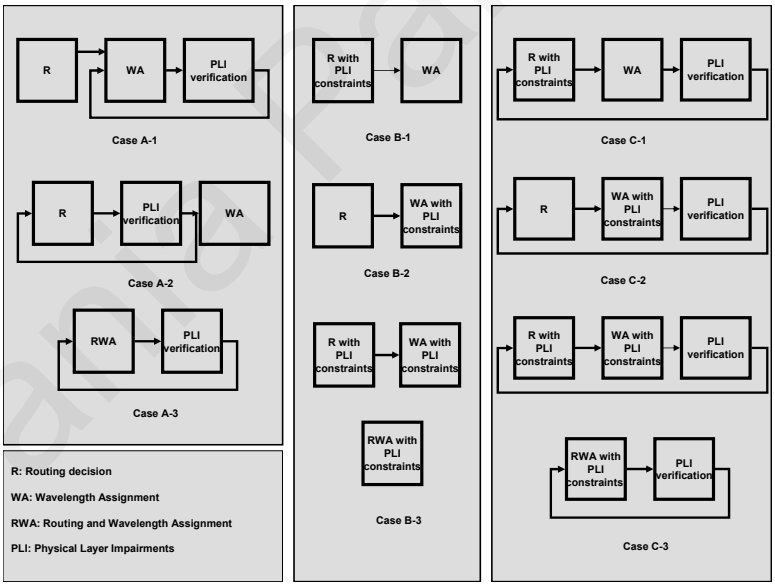

Figure 2.3: Classification of IA-RWA approaches [13]

Fig. 2.3 illustrates these cases and their various combinations as these are described in [13], while Table 2.1 provides a summary of these cases and the related surveyed papers. According to Fig. 2.3, Case A can be divided into three categories. Specifically in Case A-1, the route and the wavelength assignment are first selected without accounting for the

PLIs and then the PLIs are verified. If the PLIs are not met, only the wavelength assignment can be changed. In Case A-2, a route is first computed for the request without considering the PLIs, and then the PLIs are verified. The first two steps are performed until a route is found that meets the PLIs and then the wavelength assignment procedure is performed. Finally, in Case A-3, the RWA algorithm is performed without considering the PLIs, and then the PLI verification is executed.

Case B schemes are also divided into three categories. In general, in Case B-1 the route is computed by taking into account the PLIs and then the wavelength assignment procedure takes place. For example, during the routing procedure the link weights can represent a physical layer information such as the crosstalk, the physical distance, etc. In the B-2 scheme, the PLIs are considered only during the wavelength assignment phase, while in B-3 techniques the PLIs are considered during both the routing and wavelength assignment phases. Finally, schemes investigated for Case C are similar to the ones for Case B with the difference that in Case C approaches there is an additional phase of verification of the PLIs that enables for the lightpath selection phases to be re-attempted if needed. In the next sections, several of the heuristic and meta-heuristic algorithms of Table 2.1 that follow one of the Cases of Figure 2.3 are presented in more detail.

Table 2.1: Summary of IA-RWA Proposals and Approaches

Case	*Indicative references*
$A-1$	[12]
$A-2$	[81, 127, 162]
$A-3$	[60, 78, 113, 121, 122, 216]
$B-1$	[34, 117, 209]
$B-2$	[11, 217]
$B-3$	[33, 49, 135, 238]
$C-1$	[70, 92, 115, 126, 162, 208, 237]
$C-2$	[71, 72]
$C-3$	[114]

Note that throughout the thesis, impairment-aware (IA) provisioning of connections requests, either these are concerning unicast, multicast or groupcast connections, can follow one of the cases described above. The IA-provisioning algorithms that are followed each time are described separately in each chapter.

2.2.1 IA-RWA Heuristics

For the routing sub-problem in the fixed-routing case, a number of IA-RWA heuristics proposed in [32–34,71,72,78,127,134,144] follow the shortest path approach considering minimum number of hops for the computation of the unicast paths. Other IA-RWA algorithms can be PLI-aware by utilizing the link costs as a function of one or more PLIs. The physical distance, which is the simplest PLI-aware link cost, is used in [11,237], while other IA-RWA approaches utilized link costs expressed as a function of the residual dispersion parameter [34], the FWM crosstalk [116], the Q-factor [237], or the noise variance representing both linear and non-linear impairments [70]. Modifications to the shortest path algorithm are also proposed in [33, 113, 116] aiming at minimizing the impact of the PLIs. For example, authors in [33] proposed the Best-Optical Signal Noise Ratio (BOSNR) algorithm that jointly assigns to a given request a path and a corresponding wavelength. In particular, the path/wavelength solution that presents the maximum OSNR is selected, attempting to reduce the blocking probability caused due to unacceptable OSNR. In [116], an impairment-constraint-based shortest path algorithm that takes into account the utilization of network resources and the physical impairment due to FWM crosstalk was proposed. Specifically, the FWM-aware RWA algorithm was proposed that considers during the provisioning phase of the request the available link capacity and the level of FWM crosstalk, by updating the network state information stored in a cost function. In particular, the cost function takes into account the FWM effect and the availability of wavelengths on each link in the network. By doing so, the algorithm avoids the establishment of lightpaths that have been degraded by FWM crosstalk.

Work in [45, 60, 162, 208], that is related to the fixed-alternate routing case, utilized the hop count as the cost metric for the shortest path heuristic, while other approaches are PLI-aware by associating the link costs with the distance metric [11, 12, 126, 133] or with other physical impairments, such as PMD, ASE noise, crosstalk, CD, and filter concatenation [189]. Another approach that considers the PLIs during the routing phase is to associate the link costs with a Q-factor penalty. This cost can be based on real-time Q-factor measurements collected from devices [49] or can be calculated analytically either as the worst-case Q-factor penalty [126] or taking into account linear [92] and non-linear impairments [115]. For example, authors in [126] investigated two cases of a constraint-based routing algorithm; the off-line routing approach that is based on worst-case physical transmission penalties and the on-line approach that considers the current network state. Specifically, the algorithm

first computes k shortest paths by considering the worst-case Q-factor penalties as link costs between the source and destination nodes. Then the wavelength continuity constraint is checked and if there is no available wavelength the connection is blocked. Otherwise, the Q-factor at the destination node is calculated and if it lies above the threshold the connection is established. If not, for the off-line routing scenario, the connection request is rejected, whereas for the on-line scenario, the Q-factor is calculated once again based on the current traffic state and if the new estimation of the Q-factor is above the threshold, the connection request is set-up, otherwise it is blocked.

For the wavelength assignment sub-problem a number of approaches proposed in the literature such as [11, 12, 33, 34, 60, 68, 70, 117, 127, 132–134, 163, 208, 209] follow the First-Fit selection method in which the first non-occupied wavelength that satisfies network-layer and physical-layer constraints is selected. Other IA-RWA algorithms follow the Best-Fit [78] or the Least-Loaded [113] wavelength assignment techniques, while yet others follow a random selection approach [70, 144]. Although the First-Fit algorithm is reported to perform better compared to the random and least-loaded wavelength assignments [215], work in [144] showed that the random algorithm decreases the crosstalk effects since it tends to spread the wavelength use across the network.

For the case were the IA-RWA is solved jointly several heuristics have also been proposed [33, 121, 217]. Work in [121] proposed the A* algorithm which relies on a layered network graph that is derived from a network graph by multiplication of links and vertices by the number of corresponding and available wavelengths per fiber. This algorithm is able to find a feasible lightpath in one algorithmic step due to the layered representation of links and wavelengths. Additionally, in [217] the Minimum Crosstalk (MC) algorithm is proposed. The MC algorithm calculates the shortest path for each wavelength and also calculates the number of crosstalk components along each candidate lightpath. The lightpath with the minimum crosstalk intensity is then chosen among all the candidate lightpaths. Finally, work in [33] proposed the Best-OSNR algorithm that jointly assigns to a given request a path and a wavelength in an effort to maximize the OSNR.

2.2.2 IA-RWA Meta-heuristics

Apart from the heuristic-based algorithms, some meta-heuristic methods were also proposed in [93, 97, 113, 120, 132, 133] to solve the IA-RWA problem. Note that meta-heuristic methods, in general, allow the convergence to an optimum solution through successive iterations.

In particular, authors in [93,132,133] proposed Ant Colony Optimization (ACO) algorithms for solving the IA-RWA problem. The idea behind the ACO algorithms is based on the behavior of ants seeking a path between their colony and a source of food, aiming at finding the optimal path in a network. The ants iteratively build solutions based on their own information and on the traces left by other ants in the network nodes. For example, the ACO algorithm proposed in [93] adaptively calculates routes in the network by actively monitoring the aggregate optical power of each link. It calculates the paths on a hop-by-hop basis in the sense that the next hop is calculated based on the trace values left in the nodes that account for the Optical Performance Monitoring of the links. This algorithm calculates a multi-constraint path in a distributed manner considering the ASE and power budget constraints.

Genetic Algorithms (GA) can be also used for solving the IA-RWA problem. In general, GAs start with a set of solutions (represented by chromosomes) called population. Solutions from one population are taken and used to form a new population that is expected to be better than the old one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness; the more suitable they are, the more chances they have to reproduce. The algorithm terminates after a number of iterations when some condition is satisfied. GAs are presented in [97, 120]. Specifically, work in [97] aims at computing a lightpath in such a way that the average blocking rate and the usage of optical devices such as wavelength converters and amplifiers, is minimized. The PLIs that are considered include both the ASE and the PMD.

Finally, authors in [113] proposed a Prediction Based Routing (PBR) mechanism for solving the IA-RWA algorithm that accounts for the PLIs present by limiting the transmission distance. The main concept of the PBR mechanism is based on extending the branch prediction concepts used in the computer architecture area [175]. Bringing this concept to a network scenario, the PBR mechanism selects the lightpath between a source-destination node-pair from the behavior of previous connection requests. Thus, the PBR mechanism does not need update messages with global network state information to compute the lightpath.

2.3 Physical Layer System Model

The Bit Error Rate (BER) of the system is the main performance indicator in a fiber-optic digital communication system. However, as the BER is a difficult parameter to evaluate, the required system Q-factor for a target BER can be derived using Eq. (2.10) [1,6].

$$BER = \frac{1}{2} erfc\Big(\frac{Q}{\sqrt{2}}\Big) \approx \frac{e^{\frac{-Q}{2}}}{Q\sqrt{2\pi}} \tag{2.10}$$

Thus, the Q-factor approach is used in this thesis to account for the impairments described earlier in this chapter. In this section, the physical layer system model that is used throughout the thesis, for the calculation of the Q-factor is defined. This model is used during the provisioning phase of the requests to decide whether a connection will be admitted in the network. In particular, each time a connection request arrives into the system, the IA-RWA is solved by computing a shortest path route and wavelength assignment for that route, based on the first-fit wavelength assignment algorithm. Then the Q-factor at the destination node is calculated and the connection request is admitted into the network if its Q-factor is above a predetermined Q-threshold. If the physical layer constraints are not met, a new wavelength assignment is implemented and the heuristic is repeated until no new wavelength assignment is possible. Next, the mathematical formulation of the Q-factor is given.

2.3.1 Q-factor Formulation

The value of the Q-factor, which is a function of the wavelength λ or linearly a function of the frequency f, can be calculated using Eq. (2.11) and compared to the required performance,

$$Q(f) = \frac{I_1 - I_0}{\sigma_1 + \sigma_0} \tag{2.11}$$

where, I_0 and I_1 represent the received current levels for symbols 0 and 1 respectively and σ_i is shown in Eq. (2.12) as the sum of the variances of the thermal noise, shot noise, various components of beat noise, and RIN noise. Note that σ_0 and σ_1 denote the sum of the variances for the various noise components for symbols 0 and 1 respectively. Also note that the conversion of a wavelength $\lambda(nm)$ into the frequency $f(Hz)$ is shown in Table 2.2.

$$\sigma_i^2 = \sigma_{th}^2 + \sigma_{shot-i}^2 + \sigma_{ASE-ASE}^2 + \sigma_{s-ASE-i}^2 + \sigma_{RIN-i}^2 + \sigma_{ASE-shot}^2 \tag{2.12}$$

The above assumes a baseline system with various receiver noise terms as well as ASE noise. To include other common PLIs such as crosstalk, fiber nonlinearities, distortion due to optical filter concatenation, and PMD among others, a simple Q-budgeting approach is used as described in [6]. More precisely, incoherent common channel crosstalk penalty is budgeted at 0.8 dBQ based on a model presented in [6], the penalty due to optical filter narrowing is budgeted at 0.4 dB according to work in [6] and [54], PMD is budgeted at

0.2 dBQ based on the analytical model presented in [6] and references therein and fiber nonlinearities are factored at 1 dBQ (typical for a metro network [6]), and a safety margin of 1 dBQ is included into the budgeting model for component aging. It must also be pointed out that amplifier gain control is assumed [110] and that no polarization-dependent gain/loss (PDG/PDL) or amplifier ripple are initially present, thus precluding power instabilities. (As it will be shown later in Chapter 3, the impact of PDG/PDL will be included in the design of the network and various multicast routing techniques will be analyzed with these additional constraints.) The approach starts from the Q-value for the baseline system and budgets Q-penalties for the various PLIs present. The Q-penalty (Q_{dB}) associated with each physical layer impairment in a system is commonly expressed in dB and in this work the definition of Eq. (2.13) is used.

$$Q_{dB} = 10 \times log(Q_{linear}) \tag{2.13}$$

Thus, the Q-penalty is calculated as the Q_{dB} without the impairment in place minus the Q_{dB} with the impairment present. This approach enables a network designer to calculate the impact of physical layer effects, such as non-linear effects, polarization effects, optical crosstalk, etc., in the design of an optical network. The reader should note that when calculating the dBQ value for fiber nonlinearities (such as cross-phase modulation and four wave mixing), a worst-case value is assumed that covers the cases of varying number of channels on each path, based on work in [6] that included detailed time-domain simulations for the nonlinear effects.

The model based on the Q-factor performance of the connection is used during the provisioning phase to decide whether a request will be admitted to the network or rejected [136]. If the Q-value on the calculated lightpath is below a pre-determined threshold, then the new call will be blocked. Throughout the thesis, a Q-threshold of 8.5 dBQ is assumed, which corresponds to a BER of 10^{-12}. Note that in an optical network there are effects that are interdependent on the number of channels present at each instance in the system (optical crosstalk, cross-phase modulation, four wave mixing are some examples) and there are others that only depend on the one channel under investigation. The model used in this work allows for the case where the Q-factor constraint is applied for new calls only, with the above calculation done for these new calls. To conclude, based on this methodology, if any of these tests fail, meaning that the Q-factor for the path is below a pre-determined threshold, then the new call will be blocked.

2.3.2 Analytical Computation of the Q-factor

The analytical computation of the Q-factor is given using the receiver pre-amplifier assumptions of Fig. 2.4 and the network/component parameters of Table 2.2. In this numerical example, it is assumed that the gain (G) of the amplifier is set at 0 dB with a noise figure (NF) of 4.5 dB. The input power (P_{in}) to the amplifier is assumed to be -30 dBm, while the input ASE noise (r_{ASE}) to the amplifier is assumed to be -168.46 dBm.

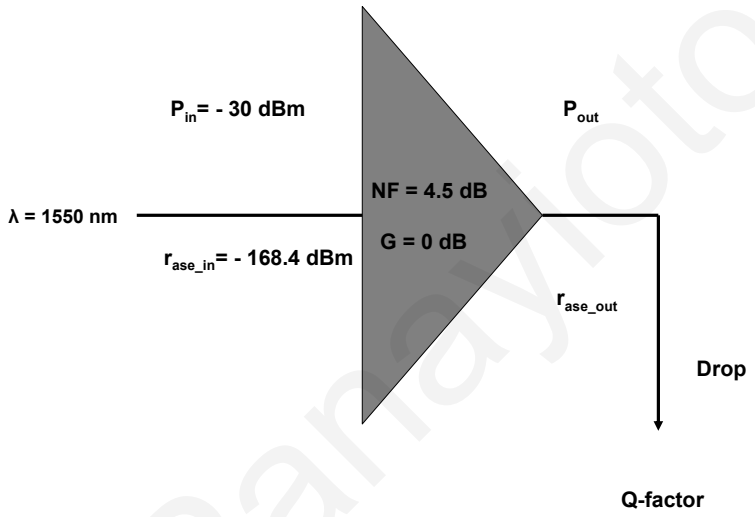

Figure 2.4: Receiver pre-amplifier parameters.

Table 2.2: Network/Component Parameters

p_1	Eye opening-level 1	1
p_0	Eye opening-level 0	0.15
e	Electron charge (Coulombs)	1.602×10^{-19}
h	Planck's constant (Joules)	6.626×10^{-34}
Bo	Optical bandwidth (GHz)	31.2
Bo_{amp}	Amplifier optical bandwidth (GHz)	12.5
Bo_{filter}	Optical filter bandwidth (GHz)	62.5
B_e	Electrical bandwidth in (GHz)	6.5
R_b	Bit rate	$B_e/0.65$
f	Frequency (Hz)	$\frac{2.99\times10^8}{\lambda(nm)\times10^{-9}}$
r	Responsivity of receiver (A/W)	1
X	Extinction ratio (dB)	10
RIN	Relative intensity noise (dB/Hz)	-145.23
NEC	Noise equivalent circuit at receiver ($pA/\sqrt{Hz}$)	14

Therefore, the ASE noise, after passing through the amplifier, is calculated according to

Eq. (2.14) resulting in a linear $r_{ASE_{out}}$ value of 1.44×10^{-17} mW which corresponds according to Eq. (2.13) to -168.39 dBm. The current that is created when the noise enters the photodiode is given by Eq. (2.15), that results to an I_{ASE} value of 4.52×10^{-7} A.

$$r_{ASE\_out} = r_{ASE\_in} \times G + h \times f \times (G \times NF - 1) \tag{2.14}$$

$$I_{ASE} = r \times r_{ASE} \times B_o \tag{2.15}$$

Since the linear gain of the amplifier is 1, the power out of the amplifier P_{out} is -30 dBm which corresponds to 0.001 mW. Hence, according to Eq. (2.16), $I_0 = 3.01^{-4}$ A and $I_1 = 0.002$ A.

$$I_i = 2 \times r \times P_{out} \times \frac{10^{\mathrm{X}/10}}{1 + 10^{\mathrm{X}/10}} \times p_i \tag{2.16}$$

The variance of the thermal noise, assuming gaussian distribution of the noise, is computed by Eq. (2.17) as $\sigma_{th} = 1.12 \times 10^{-6}$.

$$\sigma_{th} = \sqrt{NEC^2 \times Be} \tag{2.17}$$

The variance of the ASE-ASE beat noise is 2.06×10^{-7} given by Eq. (2.18), while the variance of the ASE-shot beat noise is 3.07×10^{-8} given by Eq. (2.19).

$$\sigma_{ASE-ASE} = \sqrt{(I_{ASE})^2 \times \frac{Be}{Bo}} \tag{2.18}$$

$$\sigma_{ASE-shot} = \sqrt{2 \times I_{ASE} \times e \times Be} \tag{2.19}$$

In the same way, using Eqs. 2.20, 2.21, and 2.22, $\sigma_{shot-0} = 7.91 \times 10^{-7}$, $\sigma_{shot-1} = 2.03 \times 10^{-6}$, $\sigma_{s-ASE-1} = 1.93 \times 10^{-5}$, $\sigma_{s-ASE-0} = 7.53 \times 10^{-6}$, $\sigma_{RIN-0} = 1.65 \times 10^{-6}$, and $\sigma_{RIN-1} = 1.1 \times 10^{-5}$.

$$\sigma_{shot-i} = \sqrt{2 \times I_i \times e \times Be} \tag{2.20}$$

$$\sigma_{s-ASE-i} = \sqrt{2 \times I_j \times I_{ASE} \times \frac{Be}{Bo}} \tag{2.21}$$

$$\sigma_{RIN-i} = \sqrt{10^{\frac{RIN}{10}} \times R_b \times I_i^2} \tag{2.22}$$

The sum of the variances of the thermal noise, shot noise, various components of beat noise, and RIN noise can then be calculated using Eq. (2.12). Therefore, $\sigma_1 = 2.23 \times 10^{-5}$ and $\sigma_0 = 7.83 \times 10^{-6}$ based on the values found above. The linear value of Q-factor is then 56 based on Eq. (2.11) which corresponds to a Q-factor of 17.47 dBQ based on Eq. (2.13). To include other common PLIs the Q-budgeting approach described above is then applied. Since the budgeting values that were previously described correspond to a combined value of 3.4 dBQ, then the Q-factor to the receiver is 14.07 dBQ.

2.3.3 RWA Performance Results for Unicast Connections

Even though the main body of work for this dissertation deals with the impairment-aware provisioning (routing/grooming) and protection of one-to-many connections (including multicast and groupcast applications), some performance results via simulations were initially obtained for the provisioning of unicast connections when the PLIs are considered, utilizing the physical layer model described above.

For the simulations, the NSF network topology was used with its link weights corresponding to the actual physical link distances in Km. Table 2.3 shows the statistics of the NSF network topology while Fig. 2.5 shows the actual network topology used for the simulations with modified link distances compared to the known NSF network. Unicast requests arrive into the network according to Poisson process and the holding time is exponentially distributed with a unit mean. A Q-threshold of 8.5 dBQ is assumed, corresponding to a BER of 10^{-12}. In order to determine the Q-value for each call, a baseline system Q-value is first calculated based on the signal and noise terms, assuming 10 Gbps bit rate, a preamplified p-i-n photodiode, and a WDM system with 32 wavelengths spaced at 100 GHz. In each simulation $5,000$ requests were generated for each traffic load for a total of $50,000$ point-to-point requests and the results were averaged over three simulation runs.

Table 2.3: Statistics of the NSF Network

Number of Nodes	14
Number of Links	21 (42 arcs)
Diameter	540 Km (4 hops)
Average Link Length	122 Km
Maximum Link Length	300 Km
Minimum Link Length	60 Km

Table 2.4: Noise Figure of EDFAs

Gain in dB	NF
$G < 13$	7
$13 < G \leq 15$	6.7
$15 < G \leq 17$	6.5
$17 < G \leq 20$	6
$G > 20$	5.5

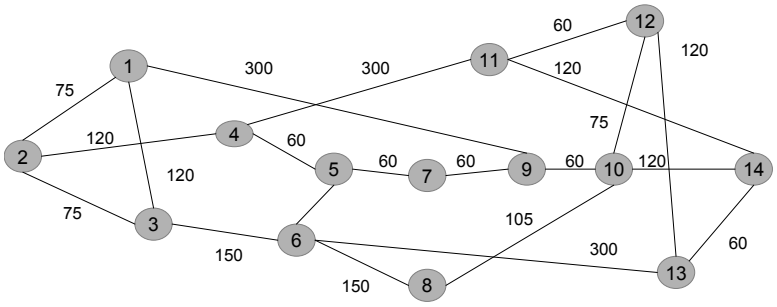

Figure 2.5: NSF network topology with scaled down distances.

For each new connection request, the RWA algorithm first solves the routing problem using Dijkstra's algorithm and then tries to assign a wavelength for that path based on the first-fit wavelength assignment technique. More precisely, the RWA algorithm attempts to assign to the path the first available wavelength. Requests are blocked if there is no available wavelength in the network. When the PLIs are also considered, requests are blocked if there is no available wavelength in the network with a Q-factor that is above the predetermined Q-threshold.

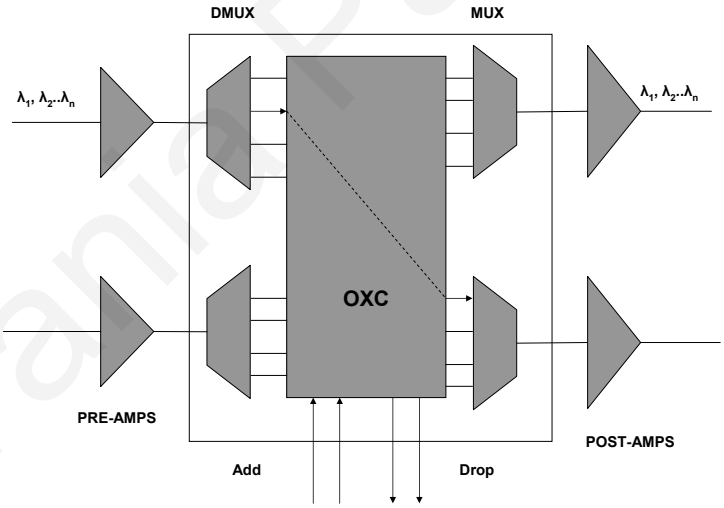

Figure 2.6: Node Architecture

The node architecture of Fig. 2.6 is assumed, in which pre-amplifiers are used for compensating for the fiber attenuation losses that are set at 0.3 dB/Km, while post-amplifiers are used for compensating for the Optical Cross-connect (OXC) losses and MUX/DMUX losses that are set at 12 dB and 1 dB respectively. DMUXs are used for demultiplexing the wave-

lengths entering the node from a specific fiber link, while the OXC is used for switching the wavelengths entering the node to the outgoing fibers. Then, MUXs are used for multiplexing the output wavelengths entering the same fiber link. Note that the NFs of the EDFAs used, are based on realistic device specifications as shown in Table 2.4 that describes how the NFs of the amplifiers are related to their gain. Note that the NF of the receiver pre-amplifier is set at 4.5 dB and the input power launched into the system is set at +3 dBm.

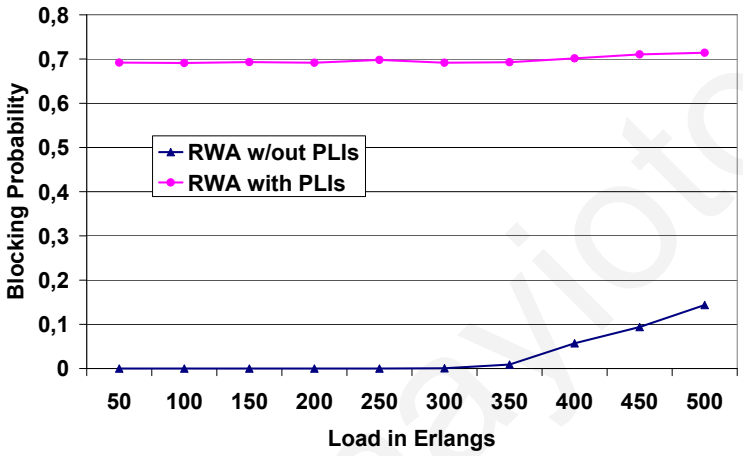

Figure 2.7: Blocking probability versus the load in Erlangs for the RWA algorithm when the PLIs are not considered and for the RWA algorithm when the PLIs are taken into account.

Figure 2.7 shows the blocking probability versus the load in Erlangs for the RWA algorithm without PLI considerations and the RWA algorithm when the PLIs are considered. Note that during the RWA algorithm that accounts for the PLIs, the physical layer constraints are not actually considered during the routing and wavelength assignment procedure of the request. There are considered only as a validation step before establishing the connection into the network. As pointed out, the classical RWA algorithm only considers the wavelength continuity and color clash constraints while when the PLIs are taken into account the Q-factor at the receivers must be above a predetermined Q-threshold. Results show that the blocking probability is significantly increased when he physical layer constraints are considered. Specifically, it shows that when the PLIs are not considered during the routing and/or wavelength assignment procedure, most of the requests that were traditionally considered feasible (RWA w/out PLIs) do not yield an acceptable Q-factor at the receiver end (RWA with PLIs); an indicator that when dealing with transparent optical networks, taking into account the various PLIs present during the routing and/or wavelength assignment procedure is of major importance.

2.4 Conclusions

In this chapter the IA-RWA problem that accounts for the physical layer constraints that degrade the signal quality in a transparent optical network was investigated. The most important PLIs were presented and a literature review of the existing unicast IA-RWA algorithms was given, including a taxonomy of various general techniques. The physical layer system model that is used throughout the thesis is subsequently presented in detail via a mathematical formulation, a computational example and through simulations that initially compared the RWA algorithm that considers the various PLIs to the conventional RWA heuristic that does not account for the PLIs present. Even though known routing and wavelength assignment heuristics are utilized for the simulations, results provide a first insight on the impact of the proposed physical layer system model to the algorithms and the system performance. Results showed the need for considering the physical layer constraints before establishing a connection into the system in order to produce solutions that can be applied in real network implementations.

Chapter 3

Multicasting in Transparent WDM Mesh Optical Networks

As discussed in the motivation of the thesis in Chapter 1, bandwidth-intensive multicast applications such as interactive distance learning, video-conferencing, distributed games, movie broadcasts from studios, etc., are becoming widely popular, mainly due to the fact that WDM optical networks can now support such applications. Multicasting has been investigated in the research community since the early days of optical networking, but has only recently received considerable attention from the service providers, mainly because now many emerging applications can potentially utilize the optical multicasting feature.

In general, multicasting refers to one-to-many connectivity and from the optical network point of view, a light-tree that spans the source and the destination set (on a specific wavelength) is used to serve the multicast request. Fig. 3.1 shows an example of a multicast tree that connects a source node s with two destinations d_1 and d_2. In most cases, there is more than one light-tree that can be used to accommodate a multicast connection, and the one with the least-cost amongst them is preferred. The calculation of these least-cost multicast trees is a fundamental problem in graph theory called the "Steiner Minimal Tree" (*SMT*) problem and it was proven to be an NP-complete problem when the multicast group has more than 2 members (at least 2 destination nodes, as the shortest path problem has a polynomial time solution) and less than the entire set of network nodes (as the spanning tree problem also has a polynomial time solution) [181]. Several ILP formulations have been proposed for designing optimum light-trees [82, 207] and several heuristics have been proposed for solving the multicast routing and wavelength assignment (MC-RWA) problem [157, 185]. However, heuristic algorithms that find "good" multicast trees (in terms of cost) and assign a wave-

length to the light-tree, are not enough to guarantee the viability of the multicast session, as when looked upon at the physical layer it may violate constraints imposed by the physical layer impairments.

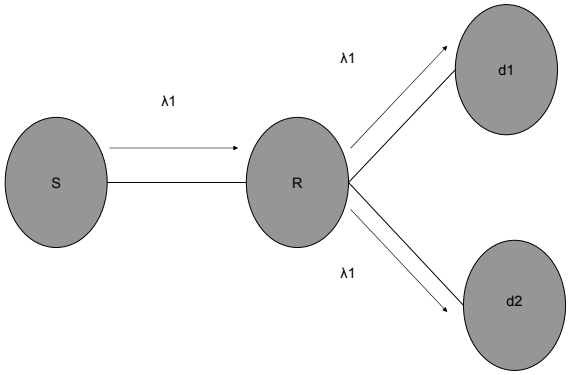

Figure 3.1: A light-tree.

In transparent optical networks, optical splitters can be used to split the incoming signal to multiple output ports thus enabling a source node to establish connections with multiple destinations. In Fig. 3.1 the signal on the incoming wavelength λ_1 splits into the two output ports of the branching node *R*. However, as optical splitters weaken the signal power, amplifiers are also required to overcome the splitting losses as well as the attenuation losses in the optical fiber. However, the optical amplifiers now also introduce noise to the optical signal (Amplified Spontaneous Emission (ASE) noise). Apart from the power loss due to the optical splitters and the ASE noise introduced by the optical amplifiers, other physical layer impairments (PLIs) discussed in Chapter 2, such as crosstalk, fiber nonlinearities, distortion due to optical filter concatenation, and polarization-mode dispersion (PMD) amongst others must be considered in the design and engineering of transparent optical networks. Although much of the existing work has been focused on the optimal placement and design of multicast-capable nodes, no work has been done so far on the design and engineering of multicast-capable nodes while taking into account the physical layer impairments. Also, the effect of different node designs and proper system engineering designs on the performance of several multicast routing heuristic algorithms has not been studied before in order to design heuristics that work well when the physical layer impairments are also taken into consideration (impairment-aware multicast routing and wavelength assignment). This is precisely what is presented in this chapter. A number of multicast-capable node architectures and system engineering designs are developed and analyzed that consider the various PLIs, followed by the design of impairment-aware multicast routing and wavelength assignment

that outperform the rest of the multicast routing and wavelength assignment techniques that were designed without accounting for the physical layer impairments. This work provides a good insight into the interaction of physical and network layers in an optical network and amply demonstrates the need for impairment-aware techniques when multicast connections need to be provisioned in transparent optical networks (TONs).

3.1 Optical Multicasting State-of-the-art

In this section first the different multicast-capable architectures proposed in the literature are presented and then the multicast routing and wavelength assignment (MC-RWA) problem along with the impairment-aware multicast-routing and wavelength assignment IA-MC-RWA problem are discussed.

3.1.1 Multicast-Capable Architectures

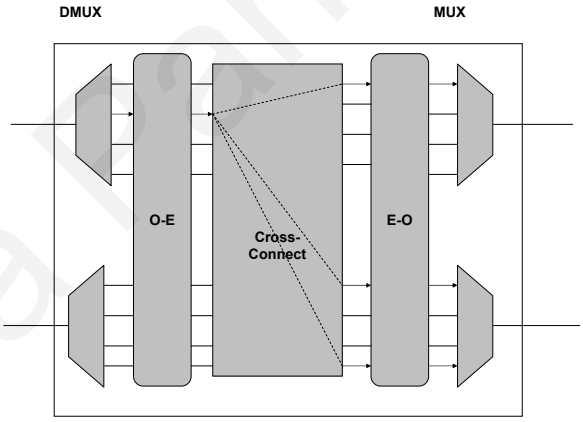

Figure 3.2: Node architecture supporting multicast connections using electronic cross-connects and OEO conversion.

To support multicast connections in WDM optical networks, multicast-capable node architectures are required to replicate the optical signal at the branching nodes of the light-trees [155]. Two approaches exist for designing nodes capable of supporting multicast traffic [169], namely the opaque and transparent approach. In Fig. 3.2 the opaque approach is illustrated, where the optical signal after being demultiplexed is converted to the electronic domain and switched using an electronic cross-connect to the appropriate output port. Then the electronic signal is converted back to the optical domain and optically multiplexed with

the other optical signals that are destined to the same output fiber. Multicasting in this case is achieved by duplicating the electronic signal prior to the EO conversion at the output ports. The opaque approach is currently very popular due to the existence of mature technology to design high-bandwidth multi-channel non-blocking electronic cross-connect switch fabrics [86].

In the transparent approach however, multicast connections can be supported more efficiently in the optical domain by utilizing the light-splitting capability of multicast-capable optical nodes rather than copying the data in the electronic domain. For multicasting in all-optical or transparent networks, optical splitters are needed to split the incoming signal to two or more outputs. An all-optical multicast-capable node, supporting n wavelengths on each link is shown in Fig. 3.3. In this figure, the optical signal from each incoming link is first demultiplexed (DMUX) into separate wavelengths and then each wavelength passes-through a multicast-capable cross-connect (MC-OXC) with splitting capabilities. The optical splitters passively split the optical signal to all (n) outgoing ports. Post-node amplifiers are required as the output signal power weakens when the input signal is split (after the optical signal passes through a splitter with n output ports, the power of the signal at each output port is only $1/n$ of the power of the input signal). Pre-amplifiers are also shown that are utilized for compensating the power losses that the signal experiences while propagating through the fiber link. A detailed design of the multicast-capable node architectures utilized in this dissertation are shown in Fig. 3.4 in Section 3.2 that follows.

As shown in [4,5,56], these power considerations affect the design of wavelength-routed networks, as well as the multicast routing heuristics [56, 201, 203]. More specifically, this dissertation examines several engineering scenarios by modifying the specifications of the different components comprising the multicast-capable node such as the noise figures (NFs) of the erbium doped fiber amplifiers (EDFAs) and their gain that compensates for the node as well as the fiber losses and by modifying the input power launched into the system. As it will be shown later in this chapter, the performance of the system is greatly affected by the engineering scenario. Furthermore, it will be shown that the performance of the multicast routing heuristics is also significantly affected by the system engineering.

In a real network, however, only some of the nodes may have splitting capabilities (sparse splitting) [112]. Also, as it is predicted that the cost associated with optical cross-connects (OXCs) and multicast-capable transparent node architectures (MC-OXCs) will still be expensive in the near future [3]] in the design of an optical network the number of nodes equipped with MC-OXCs must be taken into account. In [2, 112, 224] it was shown that an

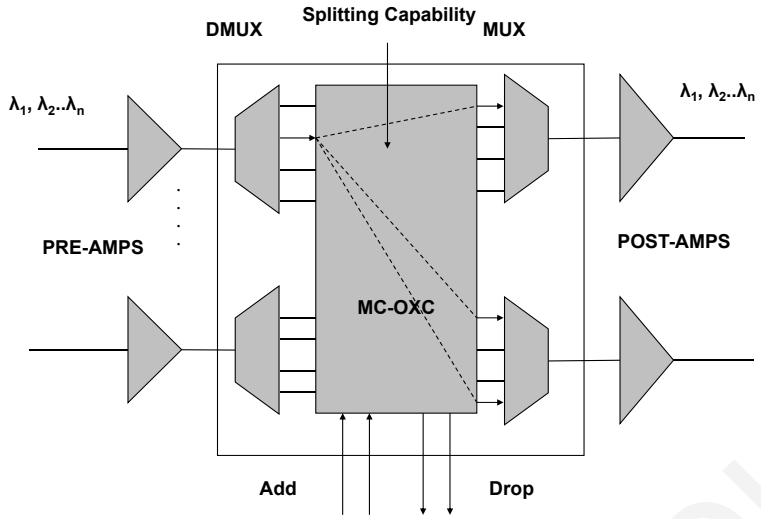

Figure 3.3: Transparent multicast-capable node architecture.

optical network with sparsely placed multicast-capable nodes can achieve performance close to that of a network where all the nodes are multicast-capable, and their optimal placement allocation was studied in [2, 3, 67, 213]. For example in [2] a heuristic approach was proposed that aims at placing in the network a fixed number of multicast-capable nodes, in such a way that the blocking probability of a set of requests is minimized. The proposed heuristic starts with a network having no multicast-capable nodes and the blocking probability of this configuration provides an upper bound for the overall blocking probability in the network. In each iteration of the heuristic, all nodes without multicast capability are chosen one at a time to be equipped with multicasting capability. The minimum blocking probability is then found amongst all possible configurations and if is less than the blocking probability found so far, the configuration that led to the current minimum blocking probability is chosen.

Apart from the optimal placement of multicast-capable nodes, in networks where only some nodes are equipped with optical splitters, the multicast routing problem must be examined. Typical multicast routing algorithms cannot be applied directly, as the physical network must been taken into consideration during the construction of the light-trees. Work in [47, 77, 77, 95, 177, 178, 191, 201, 206, 224, 228] is focused on finding multicast routing algorithms in networks with sparse splitters. For example, in [206, 224], a number of multicast routing schemes were proposed that first build an initial tree without accounting for the placement of the splitting nodes, and then the branching nodes of the tree are examined. If a branching node of the tree does not have splitting capability, its children are rerouted by avoiding branching nodes that do not have splitting capability. On the other hand, authors

in [77] proposed a multicast routing algorithm that builds the multicast tree by accounting for the placement of the splitting nodes during the construction of the tree and thus avoiding the rerouting procedure. To accomplish this, an auxiliary graph is created consisting only of the multicast-capable nodes of the original network. A link is added between two nodes in the auxiliary graph if there is a path between them in the original graph. The link weights between the nodes of the auxiliary graph represent the cost of the minimum-cost path between these two nodes in the original graph. The multicast tree is constructed by first finding a Steiner tree in the auxiliary graph and transforming it back to the original network, and then adding destinations that are not multicast-capable using a number of techniques [41, 77].

The wavelength conversion capability is also important in WDM optical networks when optical multicasting is considered. In transparent optical networks, where no wavelength converters are assumed, the same wavelength is assigned on each link along the light-tree. Hence, a signal that originates on a particular wavelength remains on the same wavelength until it reaches its destinations [131]. This is referred to as the wavelength continuity constraint. However, wavelength converters may be required in optical networks to convert the optical signal from one wavelength to another (thus reducing the number of calls blocked for not satisfying the wavelength continuity constraint). As optical wavelength converters are still expensive, difficult to implement, and not readily (commercially) available, their optimal sparse placement is studied in the literature [29, 204, 212]. For example, in [204], a heuristic algorithm is proposed that initially assumes that all the nodes in the network are multicast-capable. In each iteration of the algorithm a "wavelength convertible" cross-connect is added. First, the blocking probability of a large number of random multicast sessions is obtained on the current network configuration. Then, for each candidate node a wavelength convertible cross-connect is assumed and the blocking probability of each configuration is obtained. The candidate node that achieves the minimum blocking probability is then selected. Finally, the MC-RWA problem that minimizes the amount of wavelength converters required to support the multicast traffic is studied in [36, 199, 211, 230].

3.1.2 Impairment-Aware Multicast Routing and Wavelength Assignment Problem

In general, a multicast request is accepted into a network if a light-tree that spans the source and the destination set can be found, and if a wavelength assignment is possible for the entire light-tree. This is the well-known multicast routing and wavelength assignment problem

(MC-RWA) that as mentioned previously is proved to be NP-complete [181]. In the static case, where a set of multicast connection requests is known a priori, the MC-RWA problem is usually decomposed into two sub-problems namely the routing sub-problem and the wavelength assignment sub-problem, and then each sub-problem can be solved separately and independently. First, a multicast routing algorithm is used to compute the routes for the given set of multicast sessions, and then a wavelength assignment strategy is followed (the order that the multicast trees are assigned wavelengths can vary, depending on the heuristic algorithm used). The usual objective of the static MC-RWA problem is to maximize the multicast connection requests accepted into the network given a number of available wavelengths, or to minimize the total number of wavelengths utilized while accommodating all the multicast connection requests. Static problems are mainly considered during the design and planning of a network and not during its real-time operation considered in this thesis. A sample ILP formulation for the MC-RWA problem is shown below. The network parameters and variables used in this formulation are as follows [4]:

- C: Set of nodes in the network.
- W: Set of wavelengths
- L: Set of links in the network
- $G = (V, L)$: Undirected graph representing the network topology
- Q: Set of multicast sessions
- ψ_i: Multicast session i
- K_i: Set of alternate trees for multicast session ψ_i
- e_i:= 1 if session ψ_i is established; = 0 otherwise
- $\lambda_{i,j,c}$:= 1 if session ψ_i is established using wavelength c and the j^{th} tree; = 0 otherwise
- f_l: Number of fiber pairs on logical link l
- V_l: Set of trees with link l in their link set
- $T_{i,j}$: The j^{th} tree for session ψ_i
- σ_i: Profit of session ψ_i

$$\textit{Objective}: \qquad \textbf{Maximize } Z = \sum_{i:\psi_i \in Q} \sigma_i e_i \tag{3.1a}$$

$$\textit{subject to}: \qquad \sum_{i:\psi_i \in Q} \sum_{j:T_{i,j} \in K_i \cap V_l} \lambda_{i,j,c} \leq f_l, l \in L, c \in W \tag{3.1b}$$

$$\lambda_{i,j,c} \in \{0,1\} \tag{3.1c}$$

$$e_i \in \{0,1\} \tag{3.1d}$$

$$\sum_{j \in K_i} \sum_{c \in W} \lambda_{i,j,c} = e_i, \psi_i \in Q. \tag{3.1e}$$

The goal of this ILP formulation is to maximize the overall profit Z by choosing the appropriate tree for each session ψ_i, from the set $\{T_{i,0}, T_{i,1}, ..., T_{i,|K_i|}\}$. Constraint (3.1b) ensures that a wavelength $c \in W$ is used by at most f_l trees, constraints (3.1c) and (3.1d) force the variables to be binary, and constraint (3.1e) forces the use of at most one tree for each session. In this formulation of the MC-RWA problem, profit is maximized by establishing as many sessions as possible while choosing the most profitable ones.

In the dynamic case, where multicast requests arrive and leave the network dynamically and are not known a priori, the two sub-problems can be solved separately or jointly, as the routes can be decided according to the state of the network [73]. The dynamic MC-RWA case relates to point-and-click provisioning of multicast requests during the operation of the network and it is the case of interest in this dissertation. The usual objective in dynamic cases is to maximize the number of requests accepted into the system given a number of wavelengths, and therefore to minimize the blocking probability for the multicast connection requests. Even though the MC-RWA problem can be approximately solved by decomposing it into the routing and wavelength assignment sub-problems, each sub-problem in this case is still NP-complete.

As previously mentioned, trying to find a minimum-cost tree for a multicast connection is known as the Steiner tree problem and it was proven to be NP-complete. The wavelength assignment problem in transparent optical networks is also proved to be NP-complete as it was shown that it can be transformed to the vertex coloring problem in graph theory, a well-known NP-complete problem [40, 196, 205]. Thus, a number of heuristic algorithms were proposed for the wavelength assignment of multicast sessions such as first-fit, least-used, most-used, and random schemes as described in Chapter 2 and in [42, 184]. For example, for the wavelength assignment procedure, a layered approach can be used in networks

without wavelength conversion. Each layer, representing a copy of the WDM network, is associated with a wavelength and each layer is describing the topology of links where the wavelength is still available [52]. Work in [196] proposed two greedy wavelength assignment algorithms for dynamic all-optical WDM networks that take the network state into consideration. Both algorithms aim at minimizing the call blocking probability by maximizing the remaining network capacity after each wavelength assignment. In networks were full wavelength converters are present at every node, however, the wavelength assignment problem is not NP-complete [36, 196] and optimal solutions can be derived. Additional work in [36,137,199,211,230] was focused in finding multicast wavelength assignment algorithms that minimize the wavelength conversion cost. For example, authors in [230] proposed a wavelength assignment heuristic that first tries to assign a single wavelength to the entire tree, thus avoiding the wavelength conversion at intermediated nodes, and if this fails it divides the tree into sub-trees according to the wavelength conversion capable intermediate nodes of the tree. Then, for each sub-tree it randomly chooses a wavelength from the set of wavelengths. Furthermore, work in [231] proposed a wavelength assignment algorithm that minimizes the wavelength conversion cost in a network with limited wavelength conversion capabilities, while work in [148, 190] proposed network architectures with fixed wavelength conversion capability within the nodes of the network. Finally, work in [205] found that 25% wavelength converters are sufficient to guarantee almost the same performance with full configuration of converters at each node, even when the traffic load is extremely high.

Several heuristics have also been developed for the Steiner tree problem that take as input a graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$, and distance costs assigned to each edge $e \in E$. The output of the heuristics is a tree T spanning the set $s \cup D$. A Steiner tree heuristic approach that is also used extensively in this work is described in Algorithm 1.

Apart from the Steiner tree heuristics, another way of building the multicast tree is the Shortest path-based tree (SPT) heuristic algorithms [16, 156], that combine a number of independent shortest paths from the source to each destination node (Algorithm 2). However, SPT algorithms tend to increase the total number of links used in the tree, thus utilizing more network resources. An analytical study of the trade-offs between shortest path-based trees and Steiner-based trees can be found in [19].

A variation of the SPT heuristic, that aims at reducing the total number of links on the tree, is the Optimized Shortest Paths Tree (OSPT) heuristic [89], that enables the links that are already added to the tree to be reused by the next iteration of the algorithm. More

Algorithm 1 Steiner Tree (ST) Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$ and weights assigned to each edge $e \in E$.
Output: A tree T spanning the set $s \cup D$.

1: begin
2: $T \leftarrow s$
3: $k \leftarrow 0$
4: **while** $k < n$ **do**
5: Calculate all shortest paths from nodes $\in T$ to destination nodes $\in D$
6: Choose the shortest path amongst them and add it to tree T
7: Identify node $d_j \in D$ last added to tree T
8: Remove destination node d_j from D
9: $k \leftarrow k + 1$
10: **end while**
11: **return** T

Algorithm 2 SPT Heuristic [16, 156]

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$ and cost weights assigned to each edge $e \in E$
Output: A tree T spanning the set $s \cup D$.

1: begin
2: $T \leftarrow s$
3: **for** $i = 1 : n$ **do**
4: Find in G the minimum-cost path p_i from source node s to the multicast destination $d_i \in D$ (using Dijkstra's shortest path algorithm).
5: **end for**
6: Merge all the minimum-cost paths p_i, where $i = 1, 2, ..n$ together and construct a multicast tree
7: **return** T

specifically, when a single shortest path from the source to a destination node is calculated, the initial graph G is updated by setting a cost equal to 0 for the links that correspond to the shortest path. This way, these links have greater probability of being reused in the calculation of a next shortest path, thus reducing the total cost of the tree. The OSPT heuristic is shown in Algorithm 3.

Finally, another heuristic for building the multicast tree, is the Minimum Hop Tree (MHT) heuristic algorithm which is a modification of the ST heuristic. MHT heuristic aims at decreasing the number of hops from the source node to the destination nodes and consequently the number of total links used in the tree. The implementation of the MHT heuristic is the same as the ST heuristic, having as the only difference the input of the algorithm, where the costs on the links of the network are set to the same value (e.g., 1), instead for the actual distances of the links.

Algorithm 3 OSPT Heuristic [89]

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$ and cost weights assigned to each edge $e \in E$
Output: A tree T spanning the set $s \cup D$.

1: begin
2: $T \leftarrow s$
3: **for** $i = 1 : n$ **do**
4: Find in G the minimum-cost path p_i from source node s to the multicast destination $d_i \in D$ (using Dijkstra's shortest path algorithm).
5: Update cost= 0 for all the links in G included in the already-found path p_i
6: **end for**
7: Merge all the minimum-cost paths p_i, where $i = 1, 2, ..n$ together and construct a multicast tree
8: **return** T

Even though several heuristic algorithms exist for solving the multicast routing problem, these heuristics do not account for the PLIs encountered by the multicast connections. Furthermore, when the physical layer impairments are introduced when solving the multicast routing problem, only the power budget is typically considered. Specifically, work in [68, 69, 176, 201, 203, 213, 229] investigated the power-aware MC-RWA problem in WDM all-optical networks that considers only the power budget constraints. In those works, the objective of the MC-RWA problem was to minimize the session blocking probability and at the same time to ensure that the signal power at the destination nodes was detectable at the receivers. Authors in [68, 69] formulated this problem as a Mixed-Integer Linear Program (MILP) and also proposed a heuristic algorithm, called the power-aware multicasting (PAM) technique. Their performance results showed that the PAM algorithm produces near-optimal solutions in terms of power-budget constraints. Authors in [229] also formulated this problem as a MILP aiming at finding the light-trees with the minimum power budget while taking into account the optical power loss, node tapping loss, and light attenuation loss. A heuristic algorithm was also proposed in [201] that starts by constructing the initial tree without taking into account the power considerations and then it modifies the initial tree by replacing a set of adjacent splitters by a single splitting node, named centralized splitting node, in order to reduce the total amount of power loss. Yet another heuristic algorithm was proposed in [203], and is based on the balanced light-tree approach. Initially, the algorithm finds the multicast tree without taking into consideration the power budget constraints and then the balancing procedure takes place to minimize the power losses by decreasing the splitting losses of the light-tree. The pseudocode of balanced light-tree heuristic algorithm (BLT), as

this is developed for comparison purposes in this thesis, is shown in Algorithm 4. Finally, authors in [176] tried to maximize the number of destination nodes that could be reached transparently by investigating each lightpath on the light-trees separately and injecting different channel powers into the same optical fiber according to the distance the optical signal had to travel in order to reach the destination for each lightpath.

Algorithm 4 BLT Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$ and distance costs assigned to each edge $e \in E$ in Km

Output: A tree T spanning the set $s \cup D$ such that the difference between the maximum and the minimum split ratio of any two nodes in D is minimum.

```
1: begin
2: Find initial tree T using the Steiner Tree (ST) algorithm
3: i ← 1
4: while i > 0 do
5:    In T, calculate the split ratio of all nodes in set D
6:    Find the maximum split ratio f ∈ D
7:    u ← leaf node with maximum split ratio
8:    v ← destination node with minimum split ratio
9:    w ← the first node in the path from s to u in T such that w ∈ D or w has a fanout > 1
10:   Y ← set of nodes in path from s to v
11:   In G, compute all shortest paths from every node in set Y to node u
12:   Choose the shortest path, p_{y,u}, amongst them, where y ∈ Y and u ∈ U
13:   if max split ratio in T does not increase then
14:      Delete from T the path from w to u
15:      Add p_{v,u} to T'
16:   end if
17:   In T, calculate the split ratio of all nodes in set D
18:   Find the maximum split ratio f' ∈ D
19:   if f' < f then
20:      f ← f'
21:   else
22:      i ← 0
23:   end if
24: end while
25: return T
```

According to the above, there is a void in the research in terms of impairment-aware multicast provisioning techniques. This void is filled by the work performed in this dissertation. Different node architectures and node engineering designs, along with different multicast routing heuristic algorithms that also account for the PLIs are described analytically in this chapter using the physical layer modeling approach described in Chapter 2. Performance results for the comparison of the various architectures/engineering designs, as well as for the various impairment-aware multicast provisioning techniques are subsequently presented.

3.2 Node Architectures and Node Engineering Designs

In this section various node architecture designs are studied and analyzed. First, the performance of the network is examined for two different cases of splitting nodes, namely based on active or passive splitters, for multiple multicast routing heuristic algorithms. The main difference between the two types of splitters which are used inside each node architecture is that with passive splitters power is split as many times as the fanout of the node and controllable semiconductor optical amplifiers (SOAs) are used as gates to "cut-off" power at outputs where the signal is not destined for. Active splitters on the other hand split the power only as many times as needed for the signal to be sent to the destined outputs determined by the multicast routing algorithms.

The performance of the network is also examined for various architectures utilizing different types of transmitters and receivers. These types include all possible transmitter/receiver architecture design combinations, namely fixed transmitters/receivers, tunable transmitters/receivers, fixed transmitters/tunable receivers and tunable transmitters/fixed receivers. Different node engineering designs are also considered for the different node architecture options.

3.2.1 Passive/Active Splitter Designs

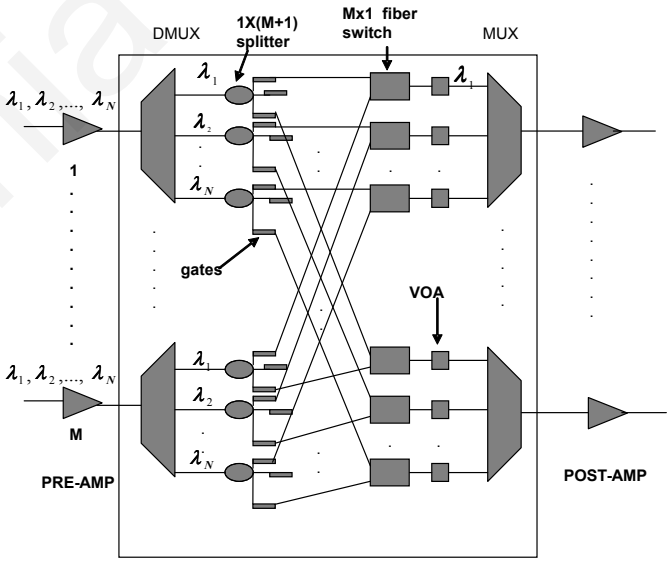

Figure 3.4: Generic transparent multicast-capable node architecture with passive splitters.

Fig. 3.4 shows a generic transparent multicast-capable node architecture with passive

splitting capabilities that is used throughout this work. In this design, N wavelengths are assumed at each input fiber and M input/output ports per node are shown. Optical multiplexers/demultiplexers (MUX/DMUXs) are used to separate the wavelengths at the input ports and then to combine them again at the output ports. Passive optical splitters are used to split the optical power to all the output ports (supporting the multicasting feature), and semiconductor optical amplifiers (SOAs) (acting as on/off gates) are utilized for preventing the optical signals from appearing at unwanted output ports. All SOAs are controlled in an intelligent manner, to avoid clashing at the same output/same wavelength of the switch. Input optical amplifiers (pre-amps), which are Erbium Doped Fiber Amplifiers (EDFAs), are typically used to compensate for the loss in the optical fiber segment preceding the node (fiber attenuation is set at 0.3 dB/Km), whereas output EDFAs (post-amps) are typically used to compensate for the losses experienced inside the node. Variable optical attenuators (VOAs) are also utilized to equalize the power prior to entering the EDFAs at the output ports for better performance and for the handling of polarization depended gain (PDG) which, although not the focus of the work presented in this section, is part of this work that will be discussed next. For simplicity, only some of the interconnection lines among the various input and output ports are shown in Fig. 3.4.

As pointed out, the case of active versus passive splitters within the node is first investigated and the network is engineered as follows: +3 dBm power is launched into each span of optical fiber, with the pre-amplifiers designed to increase power to 6 dBm coming into the optical node, and post-amplifiers are set for bringing the power back to 3 dBm as the signal is launched into the next fiber span. The above shifting of optical gain to the pre-amp improves the overall node noise figure (NF). Insertion losses for multiplexers/demultiplexers, switches, SOAs, and VOAs are based on commercially available components and some best-case values for these are shown in Table 3.1. Noise figures for the pre- and post-node EDFAs are calculated from a look-up table whose information is based on realistic commercial amplifier designs and depend on the amplifier gain taken at each time (see Table 3.2). The noise figure of the p-i-n receiver's pre-amplifier is assigned a value of 4.5 dB with a gain that is adjusted so as to bring the signal power to -4 dBm.

The insertion loss is calculated based on the worst-case scenario, considering passive or active splitters, and the amplifier gain is determined for such worst-case. This worst-case scenario is in this work limited by the maximum loss that a signal can experience passing through a given node. In particular, the determining factor is the node splitting loss, which varies depending on the node fanout and is maximized in nodes that split the signal power

Table 3.1: Typical Component Insertion Losses

Component	Losses in dB
MUX/DMUX	3
VOA	0.5
Splitter	$10 \times log(Fan\ Out)$
SOA	0.6
Switch	1

Table 3.2: Typical Noise Figure of EDFAs

Gain in dB	NF
$G < 13$	7
$13 < G \leq 15$	6.7
$15 < G \leq 17$	6.5
$17 < G \leq 20$	6
$G > 20$	5.5

the maximum amount of times. In the network used throughout this work for performance analysis purposes, the maximum fanout of any node is six, and by adding one more for local add/drop purposes the maximum power split that is used becomes seven. Note that since the end-to-end engineering of the system presented in this work depends on a worst-case scenario that depends on the maximum degree node d in the network, scalability of the network is not always possible. For example, if a new node is added into the network with a degree greater that d, then the worst-case scenario changes and the system has to be re-engineered.

Based on the above, the maximum node loss ($Node_L$) can be calculated by Eq. (3.2) as,

$$Node_L = DMUX_L + Splitter_L + SOA_L + Switch_L + VOA_L + MUX_L \tag{3.2}$$

Substituting for the loss values given in Table 3.1 and assuming the maximum splitting explained above, the total node loss in this case will be 16.6 dB. Based on this, VOAs are engineered to set the total power of each signal to a specific worst-case value so that power equalization is achieved at the input of the post-node EDFAs. It is assumed that the EDFAs have their own gain control mechanism, so that in the case when not all signals are present at each output, this ensures correct amplification. In this engineering analysis, the VOAs take into account the total power defined by Eq. (3.3) in Watts as,

$$Total\ Power = Signal\ Power + (r_{ASE} \times Bo_{filter}) \tag{3.3}$$

where r_{ASE} is the ASE noise spectral density and Bo_{filter} is a typical nominal optical bandwidth of 62.5 GHz. At the destination nodes, a pre-amplifier is assumed with noise figure of 4.5 dB and a gain such that the total input power is at -4 dBm.

3.2.2 Transmitter/Receiver Designs

Another constraint related to the physical layer is now introduced apart from the impairments consideration (through the Q-budgeting approach presented in Chapter 2), namely the transmitter/receiver availability. The modeling used in this work allows for the Q-factor calculation to be performed for any new call and in addition it examines if any available transmitters and receivers exist for that call. This means that even if the Q-factor for every path on the calculated tree is over a predetermined threshold (i.e., the path is acceptable in terms of signal quality), the new call will be blocked if no available transmitters or receivers exist for the desired wavelength and for no other alternate wavelength. Description of the node architecture and engineering design is presented next for all possible combinations of transmitters and receivers.

- *Case 1: Fixed Transmitters/Receivers:* This case assumes a fixed number of transmitters and receivers. For each source/destination node their total number is assumed to be the number of working wavelengths N times the degree of the node M. For example, if two wavelengths per fiber are used, each node will have two fixed transmitters/receivers corresponding to each input/output of the node, for a total of $2M$ transmitters/receivers at the add/drop ports (M is the node fanout). Fig. 3.5 shows a generic node architecture now consisting of passive splitters, SOAs, and fixed transmitters/receivers at the add/drop ports. For simplicity, only a few add/drop and switch interconnection lines are shown in the figure. The design shows that on the transmitting side $1 \times M$ splitters are needed for each transmitter, followed by M gates (SOAs), in order to be able to turn the signal off for any unwanted output port. On the receiving side, $N \times M$ fixed optical receivers are directly connected to the gates (SOAs) inside the node design.

- *Case 2: Fixed Transmitters/Receivers - A second Approach:* This case represents a slight variation of the aforementioned Case 1, by assuming that the number of the fixed transmitters/receivers is the number of working wavelengths in the system. This assumption relies on the fact that some of the traffic for the node will not originate/terminate there but it will be pass-through traffic (on the average for any network the passthrough traffic is approximately 75% of the total network traffic and only 25% of the network traffic is added/dropped at each individual node). The number of transmitters/receivers is now decreased to one transmitter/receiver per wavelength, and as shown in Fig. 3.6 $M \times 1$ optical switches are now needed at the receivers so as to direct

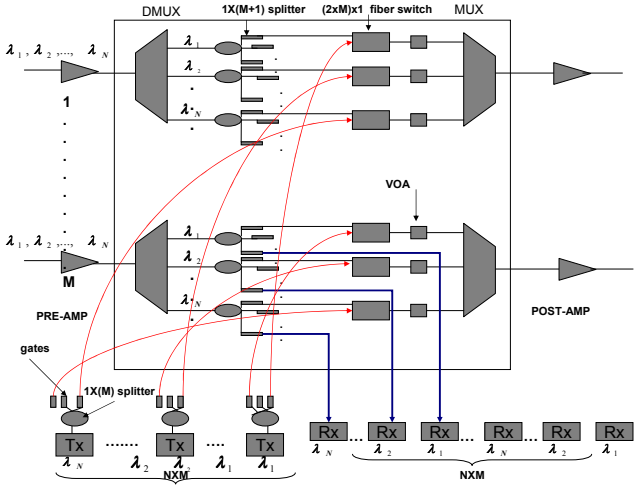

Figure 3.5: Case 1: Node engineering for fixed TXs/RXs.

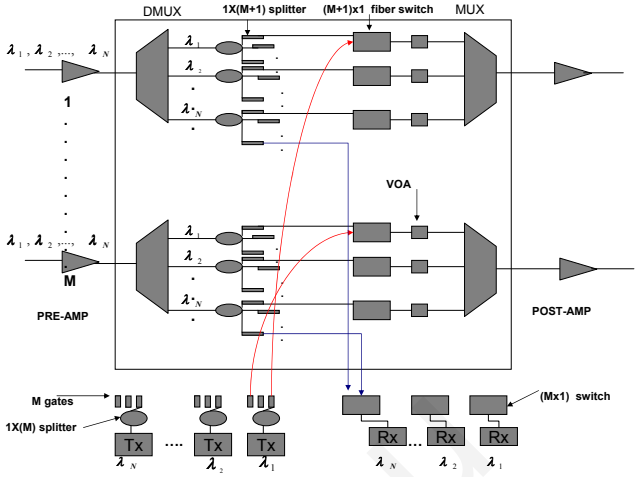

Figure 3.6: Case2: Node engineering for fixed TXs/RXs - A second approach.

the dropped wavelengths to the desired optical receiver destinations. Corresponding losses for these kind of optical switches will depend on the port size with typical values shown in Table 3.3, when these switches are implemented using MEMS technology. Again, for clarity purposes, only a few switch and add/drop interconnection lines are shown in the figure.

Table 3.3: Typical MEMS Switch Losses (switch size $K \times L$)

Size	Losses in dB
$K \times L \leq 25$	1
$25 < K \times L \leq 36$	1.5
$36 < K \times L \leq 56$	2.2
$56 < K \times L \leq 68$	3
$68 < K \times L \leq 80$	3.7
$80 < K \times L \leq 100$	4.5
$K \times L > 100$	5

- *Case 3: Tunable Transmitters/ Fixed Receivers:* This case assumes that transmitters are tunable while receivers are fixed. In both the transmit as well as the receive side of the node the available number of transmitters/receivers is equal to the number of wavelengths in the system N. As shown in Fig. 3.7, a switch is now needed at the transmit side with an add capability of fifty percent of the total number of working wavelengths. This is a pretty realistic (and even generous) percentage of wavelengths to be accessed in a system at each given add node. As a result, the dimensions of such a switch will be ($N \times M$) times ($0.5N \times M$) and its loss will depend on its size with typical losses again shown in Table 3.3. At the fixed receiver side of the node in Fig. 3.7, $M \times 1$

optical switches are needed at the receivers so as to direct the dropped wavelengths to the destined optical receiver destinations. Again, corresponding losses for these kind of optical switches will depend on the port size with typical values shown in Table 3.3, when these switches are implemented using MEMS technology. Typical losses for the rest of the components comprising the node architecture are shown in Table 3.1.

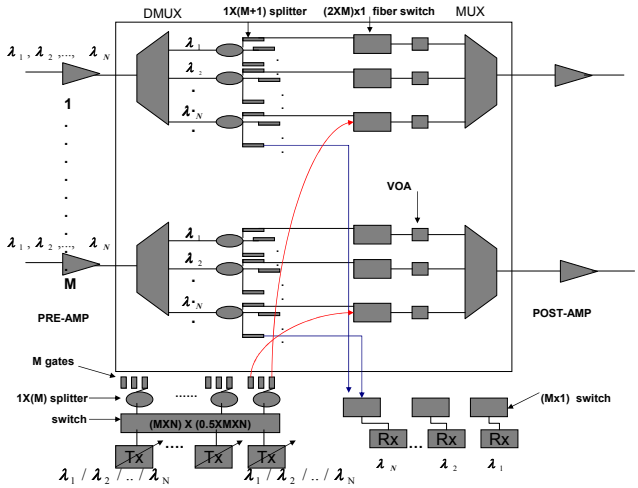

Figure 3.7: Case 3: Node engineering for tunable TXs/fixed RXs.

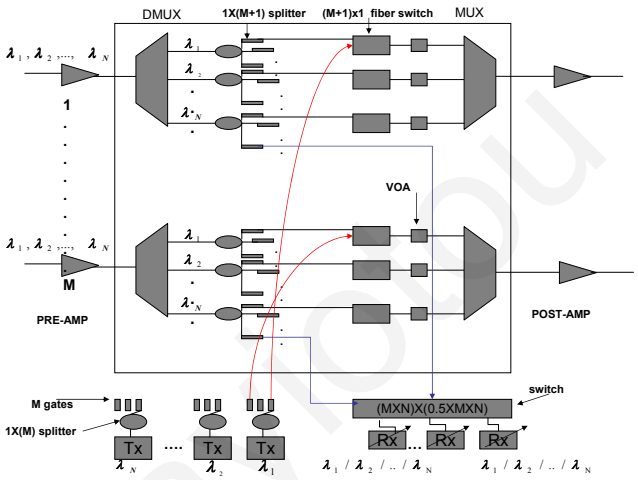

Figure 3.8: Case4: Node engineering for fixed TXs/tunable RXs.

- *Case 4: Fixed Transmitters/Tunable Receivers:* This case assumes that transmitters are fixed while receivers are tunable. In both the transmit as well as the receive side of the node the available number of transmitters/receivers is equal to the number of wavelengths in the system N. As shown in Fig. 3.8, a switch is now required at the tunable receiver side with a drop capability of fifty percent of the total number of working wavelengths (the dimensions of such a switch will again be $(N \times M)$ times $(0.5N \times M)$) and its loss will depend on its size with typical losses again shown in Table 3.3. At the transmit side, $1 \times M$ splitters are needed for each transmitter, followed by M gates, in order to be able to turn the signal off for any unwanted output port accessed by the node's transmitters.

- *Case 5: Tunable Transmitters/Tunable Receivers:* This case assumes that both transmitters as well as receivers are tunable. The number of transmitters and receivers is now equal to the number of wavelengths in the system N. As shown in Fig. 3.9, a switch is needed at both the tunable receiver as well as the tunable transmitter sides with an drop/add capability of fifty percent of the total number of working wavelengths. Again, this is a pretty realistic percentage of wavelengths to be accessed in a

system at each given add/drop node. As a result, the dimensions of such a switch will be $(N \times M)$ times $(0.5N \times M)$ and its loss will depend on its size with typical losses again shown in Table 3.3.

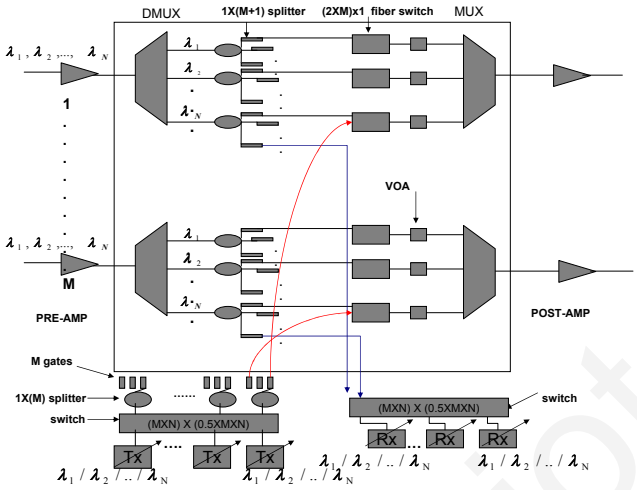

Figure 3.9: Case 5: Node engineering for tunable TXs/RXs.

For all aforementioned node architecture cases with different TX/RX designs, the node engineering is now slightly modified in order to improve the overall network performance at the receivers in terms of optical signal to noise ratio (OSNR). The optical signal launched power into the fiber is now set to +5 dBm, and each node's EDFA is assigned a realistic noise figure depending on its gain (Table 3.2), with the gain of each pre-amp compensating the loss of each preceding fiber span (0.3 dB/Km). The gain of each post-amp compensates for the actual node loss and is engineered based on the worst-case insertion loss through the node. The output powers of the pre- and post-amps are now set at +7 dBm to further improve the overall node NF. Again, the noise figure of the p-i-n receiver's pre-amplifier is assigned a value of 4.5 dB with a gain that is adjusted so as to bring the signal power to −4 dBm.

The worst-case insertion loss is now limited either by the maximum splitting loss in the case of fixed transmitters (Cases 1, 2 and 4), or by the maximum loss of the transmitter's switch in the case of tunable transmitters (Cases 3 and 5). For example, in the case of fixed transmitters, if the maximum node degree in the network is 6 and according to the aforementioned engineering, a signal originating at the maximum splitting node will reach the VOA with a power of −4.4 dBm while a signal passing-through the maximum splitting node will reach the VOA with a power of −6.05 dBm. Thus, in the case of fixed transmitters, the worst-case scenario is limited by the maximum splitting loss and the VOAs are engineered to attenuate the total power of Eq. (3.3) to −6.05 dBm. In the case of tunable transmitters, a

signal originating at the maximum splitting node will reach the VOA with a power of −9.4 dBm as the signal also experiences the insertion loss of the transmitter's switch. Specifically, when the maximum node degree is 6 and the number of wavelengths per fiber is 32, the maximum size of the transmitter's switch ($K \times L > 100$ in Table 3.3) corresponds to a maximum transmitter switch loss (TX_L) of 5 dB. On the other hand, a signal passing-through the maximum splitting node will reach the VOA with a power of −6.05 dBm based on the values of Table 3.1. Thus, in the case of tunable transmitters the worst-case scenario is limited by the insertion loss of the transmitter's switch at the maximum node degree, and the VOAs are engineered to attenuate total power of Eq. (3.3) to −9.4 dBm. Note that, for the pass-through channels, if the maximum node degree is 6, the maximum times the power is split is 7 to account for the add/drop ports while for the added channels, the maximum times the power is split is 6. According to the discussion above, the worst-case scenario in any node architecture design is given by the following equation:

$$\begin{aligned} &Node\ Loss = max\{DMUX_L + Splitting_L^P + SOA_L + Switch_L + VOA_L + MUX_L, \\ &TX_L + Splitting_L^A + SOA_L + Switch_L + VOA_L + MUX_L\} \end{aligned}$$

where $Splitting_L^P$ is the splitting loss for the pass-through channels, and $Splitting_L^A$ is the splitting loss for the add channels. In other words, the above expression returns the worst-case scenario by summing up all the losses encountered in each component in the maximum degree node in the network. The worst-case scenario can be limited either by the pass-through channels or the channels added into the network and thus both cases are examined in the above equation. The worst-case value amongst both cases is then chosen as a basis for the engineering of the network.

3.2.3 Node Engineering with Polarization-Dependent Gain/Loss Considerations

In this section, the node engineering work was expanded to include detailed modeling of polarization-dependent gain/loss (PDG/PDL) of optical components. This model is then implemented on a number of optical multicasting routing heuristics and for three specific case studies that provide the boundaries and a realistic scenario in terms of the polarization alignment of the various components in typical multicast sessions.

A statistical Monte Carlo model was used in [94] to calculate the PDL/PDG-induced

Table 3.4: Component PDG/PDL Values

Component	PDG/PDL(dB)
MUX/DMUX	0.2
VOA	0.1
Splitter	0.1
SOA gates	0.3
Switch (assume MEMs)	0.3
EDFA	0.1

channel power divergence for different sections of a worst-case path in an optical network but that work did not include any computation for the control layer multicast provisioning aspect. The latter makes the above problem computationally intractable if the statistical approach is involved. The approach in this work starts with a simple PDL/PDG model that is based on the work presented in [9]. For each amplifier, optical components, PDG and PDL models are assumed as shown in Figs. 3.10 and 3.11 (utilizing polarization beam splitters/combiners and two simple amplifier/component models, one for each max/min gain/loss polarization axis).

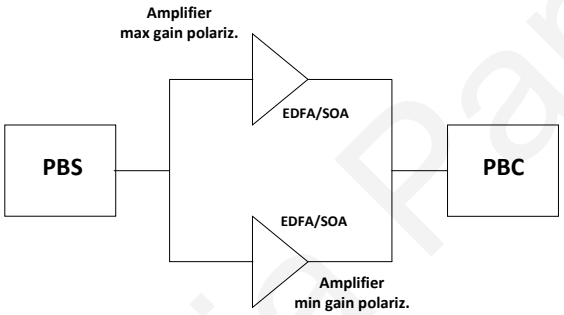

Figure 3.10: PDG model for EDFAs and SOA gates; PBS/PBC: polarization beam splitter/combiner modules.

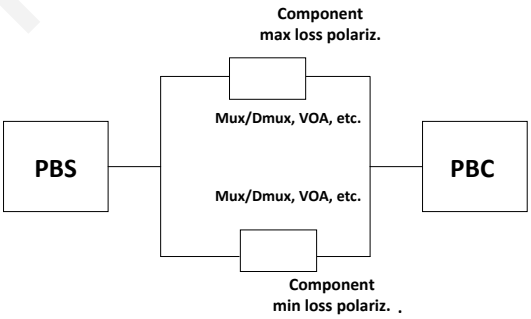

Figure 3.11: PDL model for all other optical components; PBS/PBC: polarization beam splitter/combiner modules.

The maximum gain and minimum loss of each amplifier/component are the values around which the multicast network is engineered as described in the previous sections [128, 130]. As a result, the induced PDG/PDL results in a system deviation from the above target performance point. The node architecture design shown in Fig. 3.5 is used when evaluating the impact of PDG/PDL on optical multicasting sessions, with a network engineering as previously described for the various node architectures with different TX/RX designs. The PDG and PDL values for the various components are now shown in Table 3.4 and are typical commercial values. The ASE noise being un-polarized is averaged over the two polarization axes of the model in Figs. 3.10 and 3.11.

Again, the worst-case insertion loss is calculated based on the maximum loss that a signal can experience passing through a given node. In particular, the determining factor is the node splitting loss which varies depending on the node degree and is maximized in nodes that split the signal power the maximum times. Based on this, the VOAs are engineered to set the total power of each signal to a specific worst-case value so that power equalization is achieved at the input of the post-node EDFA. The introduction of PDG/PDL is done as a perturbation to the above engineering scenario for three case studies:

- *Best-case PDG/PDL:* The assumption here is that the max-gain/min-loss polarization axes of all components are aligned with the signal.
- *Worst-case PDG/PDL:* The assumption here is that the min-gain/max-loss polarization axes of all components are aligned with the signal.
- *Random-case PDG/PDL:* A random orientation of the signal and component PDG/PDL axes is assumed over a sample of 1000 possible polarizations.

The first two case studies are meant to provide the boundaries of the performance in the multicast network and although quite unlikely in large multicast group sizes, they have been shown to be realistic for cascades of up to five amplifiers [8] (the diameter of the network used in the performance analysis consists of 13 EDFAs). The third case study assumes a more realistic randomized scenario which is computationally manageable as opposed to the methodology used in [94] which is intractable in the context of the physical/control layer interactions described here.

3.2.4 Engineering Example

Fig. 3.12 illustrates the network topology that is assumed for the numerical example described next. In this example, the node architecture and engineering design utilizing fixed TXs/RXs (Fig. 3.5) is considered, assuming that only one wavelength is present in the network. The multicast request established into the system connects source node s with destination nodes d_1, d_2, and d_3. The operating wavelength is set at $\lambda = 1550$ nm, the power launched into the system is set at $P_{in} = 5$ dBm and the fiber attenuation loss is set at $\alpha = 0.3$ dB/Km. Note that the PDL/PDG effect is not considered in this numerical example. The objective is to find the Q-factor Q_i for destination nodes d_i, where $i = 1, 2, 3$.

The different components comprising the node, such as post-amplifiers, pre-amplifiers, and VOAs, are engineered based on the worst-case scenario described above. As pointed

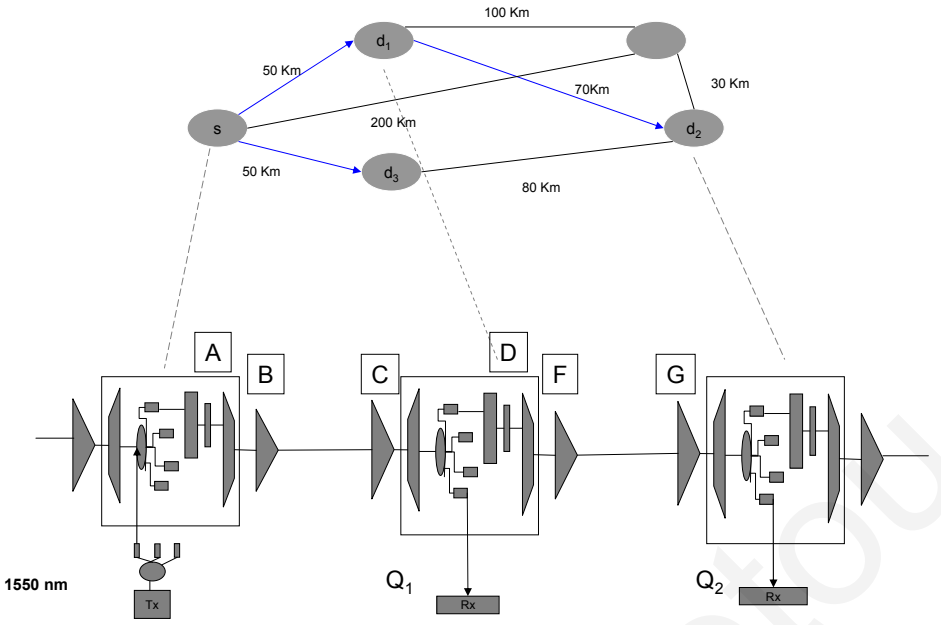

Figure 3.12: Illustrative example of Q-factor calculation using a node engineering design with fixed TXs/RXs.

out, the worst-case scenario is defined as the maximum insertion loss a signal encounters passing through the maximum degree node of the network. Considering that the maximum node degree in the network under examination is 3, then $Node_L = DMUX_L + Splitter_L + SOA_L + Switch_L + VOA_L + MUX_L = 3 + 10 \times log(4) + 0.6 + 1 + 0.5 + 3 = 14.12$ dB according to Table 3.1.

Since the target output power for both post-node and pre-node amplifiers is set at $P_{amp} = 7$ dBm, then the gain $G_{B,F}$ of the post-amplifiers must be set at 14.12 dB to compensate for the node losses. According to Table 3.2 this corresponds to a NF of 6.7 dB, and therefore $G_B = G_F = 14.12$ dB and $NF_B = NF_F = 6.7$. The gain of the pre-amplifiers depends on the attenuation loss that the signal encounters while propagating through a fiber of distance L with α being the attenuation coefficient ($G_{C,G} = \alpha \times L$).

Thus, $G_C = 0.3 \times 50 = 15$ dB with $NF_C = 6.7$, and $G_G = 0.3 \times 70 = 21$ dB with $NF_G = 5.5$. The gain of the receiver pre-amplifier is responsible for bringing the signal power at -4 dBm with $NF_{pin} = 4.5$. The VOAs which are engineered based on the worst-case scenario, are responsible for attenuating the total input power to the value of -3.62 dBm ($P_{out}^{VOA} = P_{EDFA} - DMUX_L - Splitter_L^P - SOA_L - Switch_L = 7 - 3 - 10 \times log(4) - 0.6 - 1 = -3.62$ dBm, where P_{out}^{VOA} denotes the signal power after the VOA attenuation and P_{EDFA} denotes the signal power after the amplification of the pre-amplifiers (EDFAs)).

Based on the above engineering, the signal power as well as the ASE noise dropped at

the receivers of Fig. 3.12 can be calculated. More precisely, at point A that shows the signal just before the VOA, the signal power is $P_A = -1.37$ dBm ($P_A = P_{in} - Splitter_L - SOA_L - Switch_L = 5 - 10 \times log(3) - 0.6 - 1 = -1.37$ dBm).

The total input power to the VOA is then given in mW as $P_{total} = P_A + (r_{ASE} \times Bo_{filt})$. Thus, the total power at point A is $P_{totalA} = -1.37$ dBm since no ASE noise is yet present in the system. Therefore, the VOA attenuates the signal and ASE power by $\beta = 2.25$ dB in order for the signal power to reach the preconfigured value of -3.62 dBm. Hence, the input power of the post-amplifier at point B is given by $P^B_{in} = P_A - \beta - VOA_L - MUX_L = -1.37 - 2.25 - 0.5 - 3 = -7.12$ dBm.

As $G_B = 14.12$ with $NF_B = 6.7$, the power out of the amplifier is $P^B_{out} = -7.12 + 14.12 = 7$ dBm while the amplified ASE noise as well as the ASE noise added to the existing ASE noise is given by Eq. (2.14) (Chapter 2), which corresponds to a value of -134.66 dBm.

At point C the signal reaches the pre-amplifier with a signal power of $7-(0.3\times50) = -8$ dBm and ASE noise of $-138.13 - (0.3 \times 50) = -153.13$ dBm. Both signal power and ASE noise are amplified with $G_C = 15$ dB and thus $P^C_{out} = 7$ dBm and $r^C_{ASE_{out}} = -134.66$ dBm according to Eq. (2.14). After the amplifier at point C, a portion of the signal is dropped at destination node d_1 while a portion of the signal passes-through the node. At destination d_1 the signal is dropped after the splitter and reaches the receiver pre-amplifier with a signal power of $7 - 3 - 10 \times log(4) - 0.6 = -2.62$ dBm and ASE noise of $-134.66 - 3 - 10 \times log(4) - 0.6 = -144.28$ dBm. After the amplification, the signal power at the receiver is still $P_{Rx} = -2.62$ dBm since the gain of the receiver pre-amplifier is adjusted to amplify the signal powers that are below the level of -4 dBm. The ASE noise at the receiver is calculated to be $r_{ASE_{RX}} = -144.0196$ dBm by Eq. (2.14). Then, the resultant Q-factor is $Q_1 = 14.94$ dBQ, according to the Q-factor calculation example given in Section 2.3.

Following the same steps of attenuation/amplification of the signal power and ASE noise, the Q-factor at destination node d_2 is found to be $Q_2 = 13.14$ dBQ while the Q-factor at destination node d_3 is $Q_3 = 15.02$ dBQ.

3.3 Impairment-Aware Multicast Routing Heuristic Algorithms

Several heuristic algorithms have been developed and compared in this thesis for multicast routing while also taking into consideration the PLIs. The Steiner Tree (ST) [156] as well as

the Optimized Shortest Paths Tree (OSPT) [89] heuristics that do not account for the PLIs and the Balanced Light-Tree (BLT) heuristic [203] that takes into account only the power budget constraints are also implemented for comparison purposes. In this section, three new multicast routing algorithms that consider the physical layer constraints via the Q-budgeting approach presented in Chapter 2 are described in detail. Furthermore, a new multicast routing algorithm that takes into account only the power budget constraints is also developed. The performance of each multicast routing heuristic is then compared for every node architecture and engineering scenario presented previously in this chapter.

3.3.1 Balanced Light-Tree_Q (BLT_Q) Heuristic Algorithm

Initially, the BLT_Q algorithm finds a shortest-path light-tree T using the ST heuristic, that spans the source and the destination nodes for each multicast group. This algorithm then extends the BLT approach for power budget constraints [203] by taking into account the Q-factor. Consider a light-tree, and let u denote the node with the minimum Q-factor, and denote v the node with the maximum Q-factor. The idea behind BLT_Q is to delete node u from T, and add it back to the tree by connecting it to node v in the path from source s to node v. This results in an increase of the Q-factor of node u, but it also reduces the Q-factor of all nodes below node v in the tree. Therefore, this pair of delete/add operations is performed only if it does not reduce the Q-factor of any node beyond that of node u. Thus, after each iteration of BLT_Q, the Q-factor of the node with the minimum value is increased. The algorithm also ensures that while the Q-factor of some other node(s) is decreased, it does not decrease beyond the previous minimum value. As a result, the difference between the minimum and maximum Q-factor values also decreases with each iteration. The balancing part of the algorithm terminates when two successive iterations fail to increase the minimum Q-factor. Note that if more than a pair of nodes with the same maximum and minimum Q-factor exist, we let U denote the set of nodes with the minimum Q-factor and V denote the set of nodes with the maximum Q-factor. We then select the shortest path amongst all the shortest paths that may exist between any two nodes in the sets U and V.

Figs. 3.13 and 3.14 show an illustrative example of the balancing procedure of the BLT_Q heuristic. Specifically, Fig. 3.13 shows a currently constructed tree T consisting of five destination nodes. The algorithm first identifies the destination nodes with the minimum and maximum Q-factor values. In this example, it is assumed that node d_2 is the minimum Q-factor node and d_4 is the maximum Q-factor node. Then, according to Fig. 3.14, BLT_Q

removes node d_2 from the tree, as well as all the intermediate nodes and links from node d_5 to node d_2, and adds it back to the tree by connecting it to the maximum Q-factor node d_4 via a shortest path between these two nodes. Note that, in this example, it is assumed that with the addition of node d_2 back to the tree, via the path $(d_4 - k - d_2)$, the Q-factor of d_2 is increased, while the Q-factor of the rest of the destination nodes in the tree does not decrease below the minimum Q-factor.

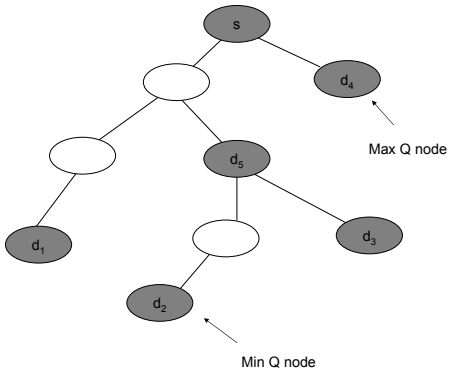

Figure 3.13: BLT_Q example: The minimum and maximum Q-factor nodes are identified on the current tree T.

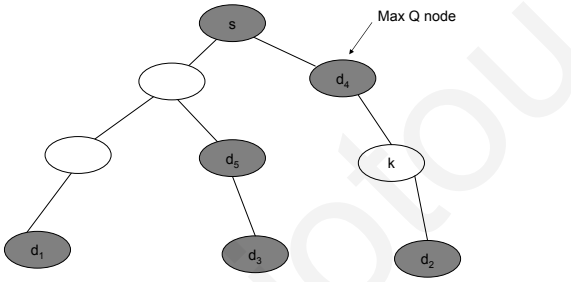

Figure 3.14: BLT_Q example: The minimum Q-factor node is deleted from the tree and added via a shortest path from the maximum Q-factor node.

The BLT_Q heuristic pseudocode is shown in Algorithm 5. As the BLT_Q algorithm tends to create trees that have more breadth than depth it decreases the attenuation loss. It also decreases the number of optical amplifiers that the signal passes through, thus decreasing the ASE noise. In contrast, even though the BLT heuristic, that accounts only for power budget constraints, creates trees that split the signal power the minimum number of times, it tends to create trees that have more depth than breadth, increasing the attenuation loss. The signals in the BLT tree, pass through a larger number of optical amplifiers thus increasing the ASE noise.

Even though BLT_Q achieves to increase the Q-factor at the destination nodes, it also increases the total number of links used in the tree, thus increasing the blocking probability due to the unavailability of resources. A variation of the BLT_Q heuristic is presented next, (BLT_$Q_{tolerance}$), that also tries to minimize the total number of the links on the tree.

3.3.2 Balanced Light-Tree_$Q_{tolerance}$ (BLT_$Q_{tolerance}$) Heuristic Algorithm

The BLT_$Q_{tolerance}$ heuristic algorithm is a modification of BLT_Q. As mentioned previously, BLT_Q increases the Q-factor of the tree nodes through the balancing procedure, but it also

Algorithm 5 BLT_Q Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$ and distance costs assigned to each edge $e \in E$ in Km.
Output: A tree T spanning the set $s \cup D$ such that the difference between the maximum and the minimum Q-factor of any two nodes in D is minimum.

```
1: begin
2: Find initial tree T using the Steiner Tree (ST) algorithm
3: Calculate the Q-factor for all destination nodes ∈ D in T
4: q ← minimum Q-factor of T
5: i ← 1
6: while i > 0 do
7:    T′ ← T
8:    U ← Set of leaf nodes with minimum Q-factor
9:    V ← Set of destination nodes with maximum Q-factor
10:   In G, compute all shortest paths from every node in set V to every node in set U
11:   Choose the shortest path, p_{v,u}, amongst them, where v ∈ V and u ∈ U
12:   w ← the first node in the path form u to s in T′ such that w ∈ D or w has a fanout > 1
13:   Delete from T′ the path from w to u
14:   Add p_{v,u} to T′
15:   Calculate the Q-factor for all destination nodes ∈ D in T′
16:   q′ ← minimum Q-factor of T′
17:   if q′ > q then
18:      T ← T′
19:      q ← q′
20:   else
21:      i ← 0
22:   end if
23: end while
24: return T
```


increases the total number of links in the tree. In order to keep the number of links on the tree low and therefore the number of wavelengths used low, BLT_Q$_{tolerance}$ is employed that is based on the minimum acceptable Q-factor. Considering that the minimum acceptable Q-factor for each path is q, this algorithm tries to maximize the Q-factor only at those destination nodes where its value is lower than q. Thus, if after a number of iterations the minimum Q-value for all destination nodes is higher than q, or if two successive iterations fail to increase the minimum Q-factor, then the balancing algorithm terminates. The pseudocode of the BLT_Q$_{tolerance}$ heuristic is shown in Algorithm 6.

Algorithm 6 BLT_$Q_{tolerance}$ Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$, distance costs assigned to each edge $e \in E$ in Km and a Q-tolerance value $q_{tolerance}$.

Output: A tree T spanning the set $s \cup D$ such that the Q-factor for every destination node $\in D$ is greater than or equal to $q_{tolerance}$.

```
1: begin
2: Find initial tree T using the ST algorithm
3: i ← 1
4: Calculate the Q-factor for all destination nodes ∈ D in T
5: q ← minimum Q-factor of T
6: while i > 0 do
7:    if q ≥ q_tolerance then
8:       i ← 0
9:    else
10:      T′ ← T
11:      U ← Set of leaf nodes with minimum Q-factor
12:      V ← Set of destination nodes with maximum Q-factor
13:      In G, compute all shortest paths from every node in set V to every node in set U
14:      Choose the shortest path, p_{v,u}, amongst them, where v ∈ V and u ∈ U
15:      w ← the first node in the path form u to s in T′ such that w ∈ D or w has a fanout
         > 1
16:      Delete from T′ the path from w to u
17:      Add p_{v,u} to T′
18:      Calculate the Q-factor for all destination nodes ∈ D in T′
19:      q′ ← minimum Q-factor of T′
20:      if q′ > q then
21:         T ← T′
22:         q ← q′
23:      else
24:         i ← 0
25:      end if
26:   end if
27: end while
28: return T
```

3.3.3 Q-based Steiner Tree (QBST) Heuristic Algorithm

The Q-based Steiner Tree heuristic algorithm aims at the construction of a minimum-cost tree in which the Q-factor at each destination node is above the predetermined Q-threshold. To achieve this, the QBST heuristic in each iteration adds to the already constructed tree T the destination node that is closer to T, only if the Q-factor of this destination node is above the predetermined Q-threshold. Otherwise it sets the links of the path that led to the current destination to a predetermined value h that is much greater compared to the link (distance) weights. The QBST heuristic terminates when every destination node is added to tree T or if

the QBST attempts to change the link weights to h but every link weight of the path that led to the current destination is already set to h. Therefore, QBST returns a multicast tree only if the Q-factor at every destination node is acceptable. The basic steps of the QBST heuristic are described in Algorithm 7.

Algorithm 7 Q-based Steiner Tree Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$, physical distance weights assigned to each edge $e \in E$, a value h several orders of magnitude greater than the average physical distance, and a Q-threshold q.
Output: A tree T spanning the set $s \cup D$ with acceptable Q-factor (greater than or equal to q) for each destination node or an indicator ($T = 0$) if such a multicast tree does not exist.

```
1: begin
2: T ← s
3: k ← 0
4: S ← s
5: while k < n do
6:    Calculate in G all shortest paths from nodes ∈ S to destination nodes ∈ D
7:    Choose the shortest path p amongst them
8:    Identify node d_j ∈ D in path p
9:    Calculate Q-factor of destination node d_j
10:   if Q ≥ q then
11:      Add p to tree T
12:      Remove destination node d_j from D
13:      Add destination node d_j to S
14:      k ← k + 1 Every link weight in p is equal to h
15:      T ← 0
16:      break
17:   else
18:      In G, change the link costs of p to h
19:   end if
20: end while
21: return T
```

3.3.4 Maximum Degree Tree (MDT_F) Heuristic Algorithm

Similar to the BLT heuristic, the MDT_F heuristic algorithm presented here does not account for the physical layer constraints apart from the power budget. It tries to control the splitting losses at the nodes by not allowing the construction of trees that have nodes with fanout greater than a predetermined value F. In this way, losses at the splitting nodes are reduced. However, this approach is appropriate only for node engineering with active splitters, since in node architectures with passive splitters losses at the splitters cannot be controlled by the algorithm but only by the node architecture. The worst-case scenario for MDT_F is based

on the maximum splitting loss that a signal encounters passing through a node, meaning that node losses are decreased. The idea behind MDT_F is to sequentially add shortest paths to T from a list of sorted shortest paths, examining whether or not the maximum degree criterion is kept. If with the addition of a new shortest path the maximum degree criterion is violated, the path is deleted and the next shortest path is selected from the list. The list of paths is sorted based on the cost of the paths in ascending order and one list per destination is calculated.

The performance of the MDT_F heuristic algorithm depends greatly on parameter F. When $F = 1$, then MDT_F creates trees with serial connected nodes, since the maximum fanout of the nodes must be one. A very small value of F may not be desirable, as only a long path is created with many hops and amplifiers. On the other hand, a large value of F creates trees that have more breadth than depth, and the total number of links in the tree is increased. It is important to note that, if a different value of F is used for each node in the network, then the algorithm can be used in optical networks with sparse splitting, since multicast-incapable nodes can be accounted for by letting $F = 1$ for these nodes. The basic steps of MDT_F heuristic algorithm are given in Algorithm 8.

3.3.5 Drop-And-Continue (DAC) Heuristic Algorithm

The DAC heuristic algorithm works similar to MDT_F when $F = 1$. The idea behind the DAC approach is the creation of a tree where the signal is not split as it passes through the nodes of the tree, unless that node is a destination. (Nodes can only perform a drop-and-continue operation now in the network.) The algorithm starts with connecting the source with its shortest destination. Then the node last added to the tree chooses a destination from the remaining destinations in set D, based on the shortest path criterion, and adds it to the tree. The same procedure is followed until all destinations are added. This approach creates trees where the nodes are connected together in a serial way, and it tends to create very long paths, decreasing the total number of links in the tree and the power loss at the nodes. However, the number of amplifiers the signal passes through is increased, particularly when the size of the multicast group is large compared to the total number of nodes in the network. The pseudocode of MDT_F is slightly changed in Algorithm 9 for the DAC heuristic algorithm.

Algorithm 8 MDT_F Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \subseteq V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$ and distance costs assigned to each edge $e \in E$ in Km.
Output: A tree T spanning the set $s \cup D$ such that every node $\in T$ has a fanout of at most F. If no such a tree exists, the algorithm returns $T = \emptyset$.

```
1: begin
2: S ← s
3: T ← s
4: k ← n
5: while k > 0 do
6:    Find set P consisting of all shortest paths from node(s) ∈ S to destination node(s) ∈ D.
7:    Sort shortest paths in P based on their cost, with the shortest path amongst them placed
      first on the list.
8:    Let l be the number of shortest paths in P.
9:    i ← 1
10:   while i ≤ l do
11:      Identify shortest path p_i ∈ P.
12:      if With the addition of p_i to T the fanout of nodes in T does not exceed F then
13:         Add p_i to T
14:         Identify destination node d_j last added to T.
15:         Remove destination node d_j from D.
16:         Add destination node d_j to S.
17:      else
18:         i ← i + 1
19:      end if
20:   end while
21:   k ← k − 1
22: end while
23: return T
```

3.4 Impairment-Aware Provisioning of Multicast Connections

In this section three impairment-aware multicast routing and wavelength assignment (IA-MC-RWA) algorithms are presented. For the first two approaches the validation of the PLIs is performed after the computation of the light-trees (in the first scheme, during the provisioning phase of the requests, only the wavelength continuity constraint must be met, whereas in the second case the availability of transmitters and receivers is also examined). The third IA-MC-RWA algorithm, namely the decomposed IA-MC-RWA algorithm, differs from the other two in the sense that apart from the TXs/RXs consideration, it aims at improving the session blocking probability by breaking each multicast tree into several sub-light-trees according to a predetermined Q-threshold. The validation of the PLIs is not required here, as

Algorithm 9 DAC Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$ and distance costs assigned to each edge $e \in E$ in Km.
Output: A tree T spanning the set $s \cup D$ such that every node $\in T$ has a fanout of at most 1.

```
1: begin
2: T ← s
3: k ← 0
4: v ← s
5: while k < n do
6:     Find set P consisting of all shortest paths from node v to destination node(s) ∈ D.
7:     Sort shortest paths in P based on their cost, with the shortest path amongst them placed
       first on the list.
8:     Identify shortest path p among the paths in P.
9:     Add p to T
10:    Identify destination node d_j last added to T.
11:    Remove destination node d_j from D.
12:    v ← d_j
13:    k ← k + 1
14: end while
15: return T
```

only light-trees with a Q-value above the predetermined Q-threshold can be created.

3.4.1 IA-MC-RWA Algorithm without TX/RX Considerations

For each multicast request that randomly arrives, the impairment-aware multicast routing and wavelength assignment (IA-MC-RWA) algorithm first solves the routing problem by finding a multicast tree that can accommodate the request and then tries to assign a wavelength for the tree based on the *first-fit* wavelength assignment technique [185]. Multicast requests are blocked if there is no available wavelength for the entire tree. If a wavelength assignment is possible, the Q-factor for each path on the tree is evaluated and the multicast request is blocked if there is at least one route on that tree with a Q-value that falls below a predetermined threshold value and there is no alternate wavelength assignment possible. Otherwise, a new wavelength assignment is implemented and the heuristic is repeated. Fig. 3.15 shows the flowchart describing the aforementioned IA-MC-RWA methodology.

3.4.2 IA-MC-RWA Algorithm with TXs/RXs Considerations

When the transmitters/receivers constraint is also included in the impairment-aware provisioning of multicast connections, the IA-MC-RWA algorithm is slightly modified to account for this constraint. Again, for each multicast request that randomly arrives, the IA-MC-RWA

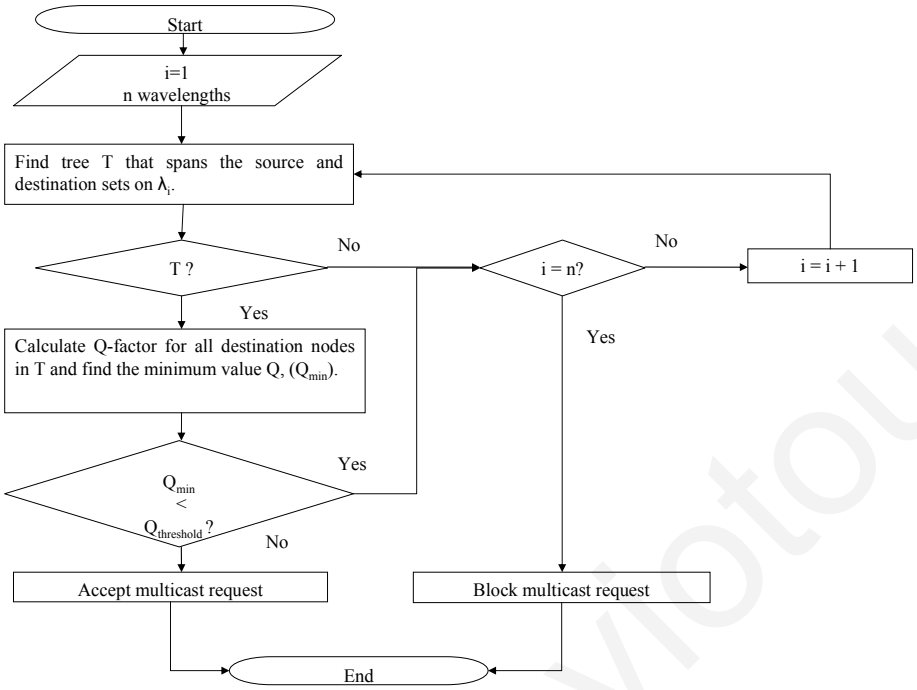

Figure 3.15: Flowchart of the IA-MC-RWA algorithm without TX/RX considerations.

algorithm first solves the routing problem by finding a multicast tree that can accommodate the request and then tries to assign a wavelength for that tree based on the *first-fit* wavelength assignment technique. Unlike the IA-MC-RWA of Fig. 3.15, the modified IA-MC-RWA algorithm attempts to assign to the tree the first available wavelength in terms of availability of links and availability of transmitters and receivers. Multicast requests are blocked if there is no available wavelength for the entire tree either due to the transmitters/receivers constraint or due to the unavailability of links. If a wavelength assignment is possible, the Q-factor for each path on the tree is evaluated and the multicast request is blocked if there is at least one route on that tree with a Q-value that falls below a predetermined threshold value and there is no alternate wavelength assignment possible. Otherwise, a new wavelength assignment is implemented and the heuristic is repeated. Fig. 3.16 shows the flowchart describing the IA-MC-RWA methodology when the transmitters/receivers constraint is also included.

3.4.3 Decomposed IA-MC-RWA Algorithm with TXs/RXs Considerations

The key-idea of the Decomposed IA-MC-RWA (D-IA-MC-RWA) algorithm is to support each multicast request with a sufficient number of light-trees in order to achieve for each

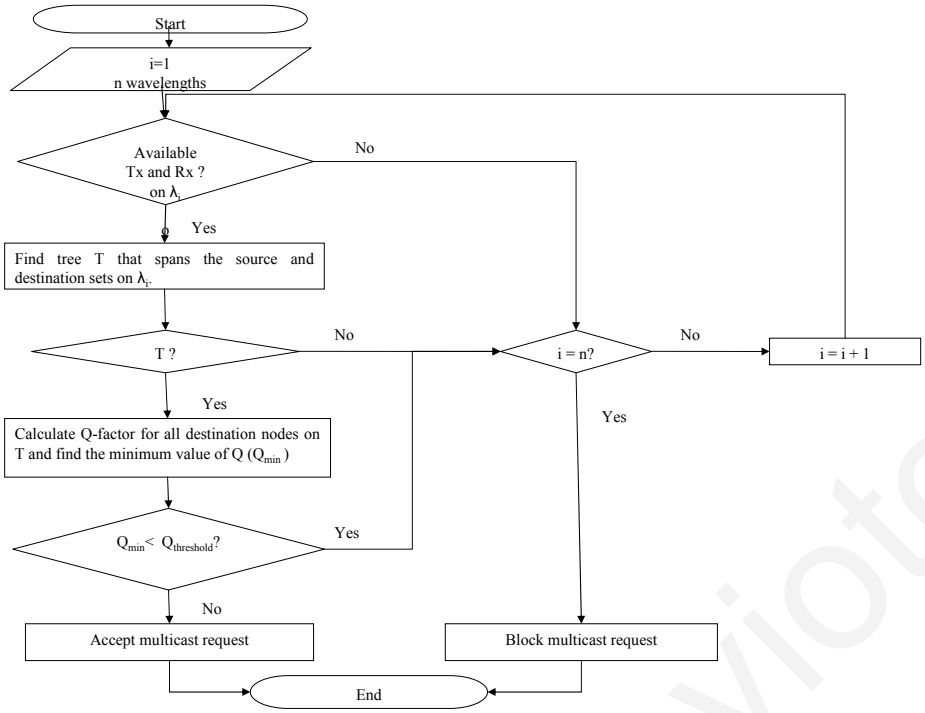

Figure 3.16: Flowchart of the IA-MC-RWA algorithm. The available transmitters/receivers constraint is also included.

destination node in the multicast set a Q-value that is above the predetermined Q-threshold. Specifically, once a multicast request arrives into the network, the D-IA-MC-RWA algorithm builds a light-tree supporting only those destination nodes that result in a Q-value that is above the predetermined Q-threshold. The routing sub-problem can be solved according to any multicast routing algorithm, while the wavelength assignment sub-problem is solved according to the *first-fit* heuristic. If some destination nodes are not included in the previous light-tree(s), due to unacceptable Q-factor, the D-IA-MC-RWA algorithm moves to the next available wavelength and computes a new light-tree spanning the source and each of the remaining destination node. The heuristic is repeated until every destination node is supported by a light-tree or until there is no available wavelength to support the remaining destinations. Multicast requests are blocked if there is no wavelength assignment for at least one of the destination nodes in the multicast set. If all destination nodes are included in at least one light-tree, and there are available TXs and RXs to support each light-tree, then the multicast request is accepted.

Even though the decomposition of multicast sets into several light-trees decreases the number of calls blocked due to a low Q-factor, more than one transmitters may be needed for just a single request, as the same information may have to be transmitted on multiple

wavelengths. Therefore, compared to the case were a single wavelength was used to serve an entire multicast call (IA-MC-RWA), the blocking probability due to low Q-factor is reduced in the expense of a higher blocking probability due to the unavailability of transmitters. The detailed pseudocode of the D-IA-MC-RWA algorithm is given in Algorithm 10.

Algorithm 10 D-IA-MC-RWA

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$, weights assigned to each edge $e \in E$ in Km, a list of wavelengths $W = [w_1, w_2, ..., w_k]$ and a Q-threshold q.
Output: A set of light-trees T originating at node s and spanning the set of nodes D.

```
1: begin
2: i ← 1
3: while i ≤ k do
4:    Find light-tree T_i originating at source node s and spanning the set D, according to
      any known multicast routing heuristic.
5:    if no such a light-tree exists on w_i and i = k then
6:       Block Multicast Request.
7:       else if no such a light-tree exists on w_i then
8:       i ← i + 1
9:    else
10:      Calculate the Q-values for each destination node ∈ D.
11:      Identify set of destination nodes D′ with a Q-value above q.
12:      Remove nodes D′ from set D.
13:      In T_i, identify lightpaths p_d that connect source node s with every destination node
         ∈ D′.
14:      Create light-tree T′_i by merging p_d lightpaths.
15:      Add T′_i into set of light-trees T.
16:      if i = k and D set is not empty then
17:         Block Multicast Request.
18:         else if i < k and D is empty then
19:         i ← k + 1
20:      else
21:         i ← i + 1
22:      end if
23:   end if
24: end while
25: if There are available TXs/RXs for each light-tree ∈ T then
26:    Accept Multicast Request.
27:    return T
28: else
29:    Block Multicast Request
30: end if
```

3.5 Complexity Analysis of Multicast Routing Heuristic Algorithms

The computational complexity of the multicast routing heuristics presented in this chapter is evaluated in this section. For every heuristic algorithm it is assumed that n is the number of nodes in a given network, e is the number of links in the given network and k is the number of destination nodes for a given multicast call. According to [51], the complexity of Dijkstra's algorithm is $n(n+1)/2 \in O(n^2)$, and according to [187] the complexity of the ST heuristic is $O(kn^2)$.

- *BLT heuristic:* According to [203] the complexity of BLT is $O(In^2)$ where I is the number of iterations of the balancing procedure of the BLT heuristic. Note that BLT is repeated until two successive iterations fail to reduce the maximum split ratio and that BLT in each iteration tries to decrease the split ratio of a leaf node in the tree. Leaf nodes are always destination nodes and therefore, the worst-case scenario is caused in the case where BLT performs $k-1$ iterations and in each iteration, succeeds to decrease the splitting ratio of a single destination node, without increasing the splitting the ratio of any other destination node to the tree. Note that k is the number of destination nodes in the multicast set. Thus, $I = k - 1$ and since in each iteration Dijkstra's algorithm is executed, then the complexity of BLT is $(k-1)n(n+1)/2 \in O(kn^2)$.

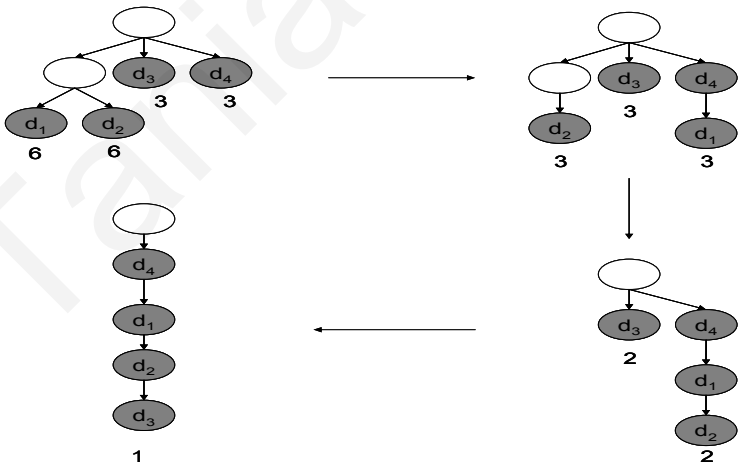

Figure 3.17: Example of the BLT heuristic, with 4 destination nodes and 3 iterations.

Fig. 3.17 shows an example of the BLT heuristic, with 4 destination nodes. Note that in Fig. 3.17 the algorithm terminates when the splitting ratio of every destination node becomes one, which means that no other reduction on the splitting ratio can be

performed. Also note that while destination nodes can be deleted and added back to the tree in a different order, or they can be connected on an alternate node than the one shown in the example, that could cause a termination of the algorithm before the worst-case number of iterations (in this example this number is 3 $(k-1)$) . The above example can also be expanded for a larger number of destination nodes.

- *BLT_Q heuristic:* In the BLT_Q heuristic, the balancing procedure is repeated until two successive iterations fail to increase the minimum Q-factor. BLT_Q in each iteration tries to decrease the Q-factor of a leaf node in the tree. Leaf nodes are always the destination nodes with the minimum Q-factor. Thus, the worst-case scenario is caused in the case where BLT_Q performs $k-1$ iterations and in each iteration succeeds to increase the Q-factor of a single destination node by connecting it to the destination node with the maximum Q-factor. Since in each iteration Dijkstra's algorithm is used, the complexity of BLT_Q is $(k-1)n(n+1)/2 \in O(kn^2)$.

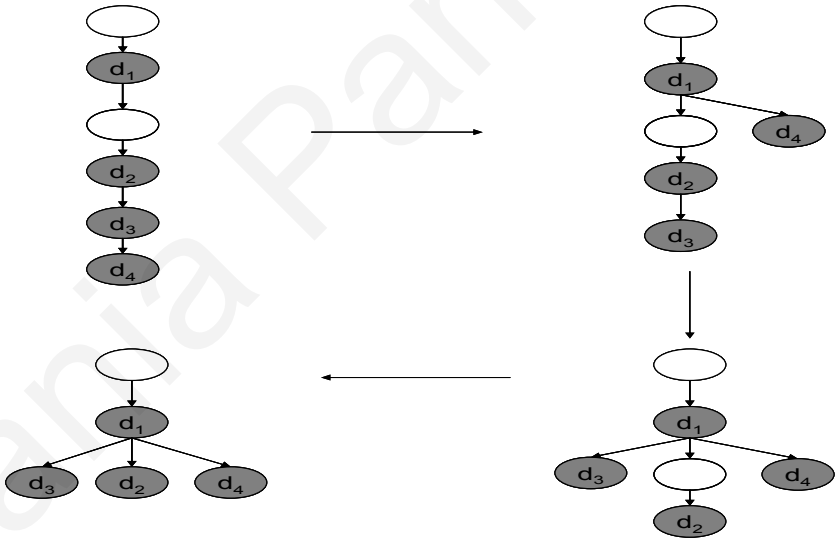

Figure 3.18: Example of the BLT_Q heuristic, with 4 destination nodes and 3 iterations. Links shown could be physical or logical.

Fig. 3.18 shows an example with 4 destinations in which 3 iterations are performed. Note that in Fig. 3.18, the algorithm terminates when the Q-factor of every destination cannot be increased any further (each destination node is pushed as close to the source node as possible).

- *BLT_$Q_{tolerance}$ heuristic:* BLT_$Q_{tolerance}$ differs from BLT_Q as BLT_$Q_{tolerance}$ improves

the Q-factor only at those destination nodes where the Q-factor is below the predetermined Q-threshold. If we assume that the worst-case scenario is caused when every destination node has a Q-factor below the predetermined Q-threshold, and BLT_Q$_{tolerance}$ achieves to increase the Q-factor at every destination node (one node in each iteration), then the complexity of BLT_Q$_{tolerance}$ is equal to the complexity of BLT_Q, that is $O(kn^2)$.

- *QBST heuristic:* In each iteration of the QBST algorithm, a destination node may be added to the currently constructed tree or not. A destination node is added if its Q-factor is above the predetermined Q-threshold. Otherwise, the costs of the links that led to that destination node are modified and the algorithm is repeated for the destination node that is now closer to the currently constructed tree. If we assume that eventually all destination nodes are added to the tree then Dijkstra's algorithm is repeated at least k times. However, it is possible that a destination will be not added from the first trial. In the worst case scenario only one link changes its cost in each failed trial, thus the number of failed trials in the worst case corresponds to the number of links in the network, e. Thus, the complexity of QBST heuristic is given by $(k + e)n(n + 1)/2 \in O((k + e)n^2)$.

- *MDT_F heuristic:* In each iteration of the MDT_F heuristic a destination node is added to the currently constructed tree provided that the maximum degree constraint is not violated. If we assume that MDT_F achieves to add, in each iteration, a destination node to the tree using Dijkstra's algorithm, then the complexity of MDT_F is given by $kn(n + 1)/2 \in O(kn^2)$.

- *DAC heuristic:* The DAC heuristic is repeated k times and each time, Dijkstra's algorithm is executed. Thus, the complexity of DAC is given by $kn(n + 1)/2 \in O(kn^2)$.

Table 3.5 summarizes the complexity of each multicast routing heuristic. From all the routing techniques, the computational complexity of the QBST heuristic is the worst as it increases not only with the multicast group size but also with the number of links in the network. The complexities of the other multicast routing heuristics are equal, but it must be noted that for the BTL, BLT_Q, and BLT_Q$_{tolerance}$ heuristic algorithms the complexity of the ST heuristic must also be added, since before the balancing procedure an initial tree must first be computed.

Table 3.5: Complexity of Multicast Routing Heuristic Algorithms

Algorithm	**Complexity**
ST	$O(kn^2)$
BLT	$O(kn^2)$
BLT_Q	$O(kn^2)$
BLT_$Q_{tolerance}$	$O(kn^2)$
QBST	$O((k+e)n^2)$
MDT_F	$O(kn^2)$
DAC	$O(kn^2)$

3.6 Performance Results

In order to evaluate the average performance of the heuristic algorithms for different network designs/system engineering parameters, multicast connections are simulated on a metro network consisting of 50 nodes and 98 links (196 arcs), with an average nodal degree of 3.92 and an average distance between the nodes of 60 Km. The maximum node degree of the network is 6 and the diameter of the network is 305 Km (6 hops). Fig. 3.19 illustrates the topology of the metropolitan area network used for the simulations while Table 3.6 describes the distance between the network nodes. Note that this table shows a list of nodes in the first column that are connected to the set of nodes shown in the second column. In the third column the distances between each set of nodes is denoted in Km. For example, according to Table 3.6, node 1 is connected to nodes 2, 5, 12, 25, and 50, and the link distances in Km are $(1,2) = 30$, $(1,5) = 60$, $(1,12) = 59$, $(1,25) = 93$, and $(1,50) = 30$. Also note that each link in the table is shown only in one direction but it is assumed that both directions are available in the network (same distances for each direction). The metropolitan area network described above is utilized throughout the rest of the thesis while MATLAB is the tool that was used for the evaluation of the simulations results.

A dynamic traffic model is used where multicast sessions arrive at each node according to a Poisson process and the holding time is exponentially distributed with a unit mean. In this work, a Q-threshold of 8.5 dBQ is assumed, corresponding to a BER of 10^{-12}. In order to determine the Q-value for each call, a baseline system Q-value is first calculated based on the signal and noise terms, assuming 10 Gbps bit rate, a pre-amplified p-i-n photodiode, and a WDM system with 32 wavelengths spaced at 100 GHz. Externally modulated transmitters and standard NRZ modulation is assumed. In each simulation $5,000$ requests were generated for each multicast group size for a total of $40,000$ multicast requests and the results were averaged over five simulation runs. The blocking probability was calculated for each simulation run while varying the multicast group size for a network load of 100 Erlangs.

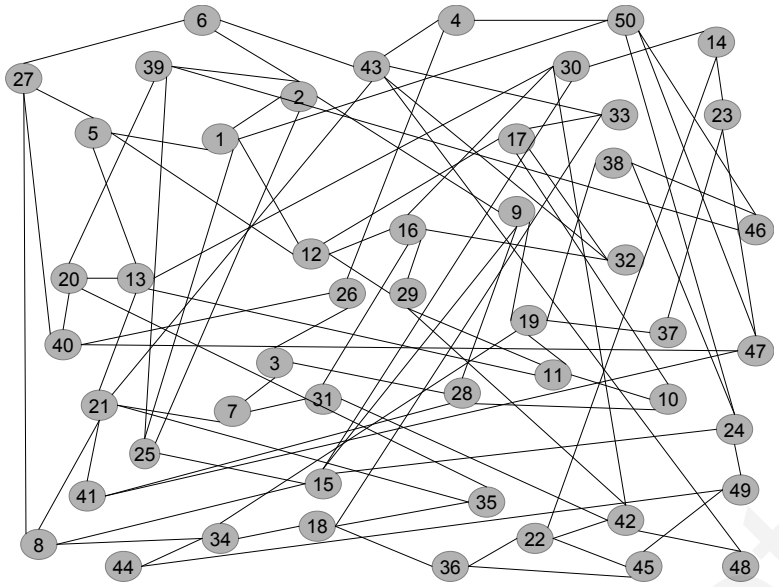

Figure 3.19: Metropolitan area network used in the simulations.

The first set of simulations deals with the question of what type of splitters should be used in the optical nodes, namely active or passive, while the next set of figures presents simulation results for the blocking probability versus the multicast group size when a number of multicast routing heuristic algorithms are used assuming different node engineering scenarios. Finally, the last set of figures focuses on the impact of PDG and PDL on the algorithms and the system performance.

3.6.1 Passive vs. Active Splitting

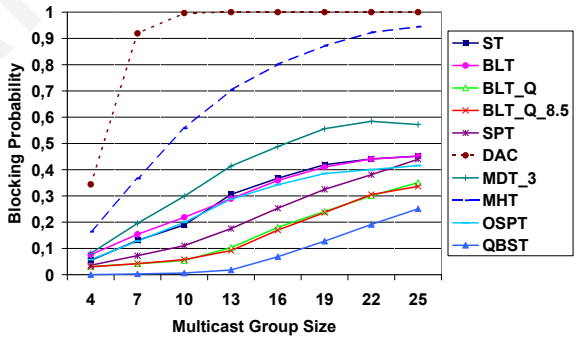

Figure 3.20: Blocking probability versus multicast group size for node engineering with active splitters.

Fig. 3.20 shows the blocking probability versus the multicast group size for node engineering with active splitters for a number of multicast routing heuristics. The IA-MC-RWA

Table 3.6: Metro Network Representation

Node	Connected to Node	Distance in Km
1	2, 5, 12, 25, 50	30, 60, 59, 93, 30
2	6, 9, 35, 39	35, 70, 35, 90
3	7, 26, 28	58, 64, 68
4	26, 43, 50	40, 35, 100
5	12, 13, 27	50, 40, 35
6	27, 43	64, 84
7	21, 32	66, 55
8	15, 21, 34, 77	40, 55, 40, 77
9	15, 19, 28	25, 82, 95
10	11, 17, 28	84, 83, 40
11	13, 19, 29	50, 84, 82
12	16, 17, 29	74, 62, 35
13	20, 21, 30	74, 82, 94
14	22, 23, 73	95, 40, 73
15	24, 25, 31	72, 40, 85
16	29, 30, 31, 32	94, 30, 84, 81
17	32, 32	40, 65
18	33, 34, 35, 36	74, 40, 94, 32
19	34, 37, 38	32, 65, 30
20	35, 39, 40	40, 82, 30
21	35, 41, 43	64, 40, 58
22	36, 42, 45	72, 28, 35
23	37, 47, 48	75, 64, 53
24	38, 49, 50	40, 75, 40
25	39	62
26	40	65
27	40	30
28	41	62
29	42	20
30	42	74
31	42	62
32	53	65
33	43	40
34	44	72
35	44	40
36	45	30
37	45	40
38	46	85
39	46	74
40	47	40
41	47	20
42	48	84
43	48	87
44	49	98
45	49	76
46	50	83
47	50	40

algorithm used for the simulations is shown in Fig. 3.15. Results show that the QBST heuristic algorithm that takes into account the PLIs performs the best compared to the other multicast routing algorithms that either consider only the wavelength continuity constraint or also take into account the power budget constraints. The second-best algorithms are both the

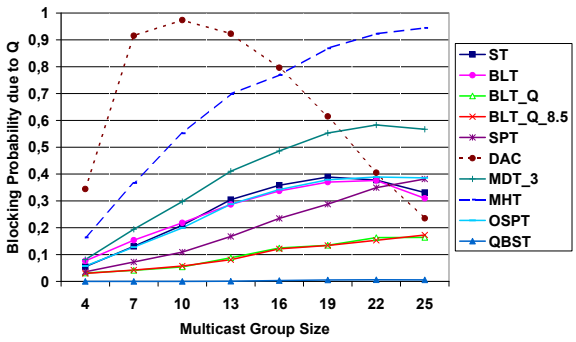

Figure 3.21: Blocking probability due to Q versus multicast group size for node engineering with active splitters.

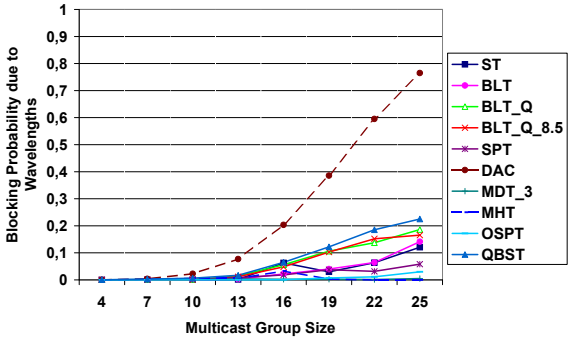

Figure 3.22: Blocking probability due to the unavailability of wavelengths versus multicast group size for node engineering with active splitters.

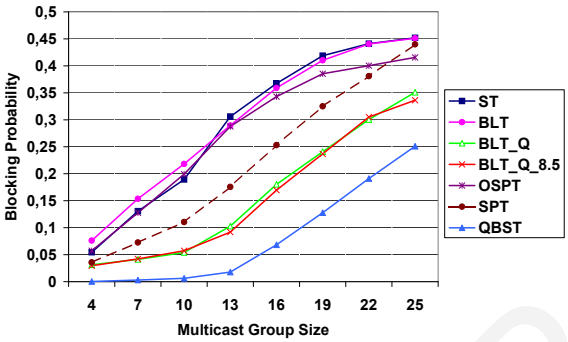

Figure 3.23: Blocking probability versus multicast group size for node engineering with active splitters.

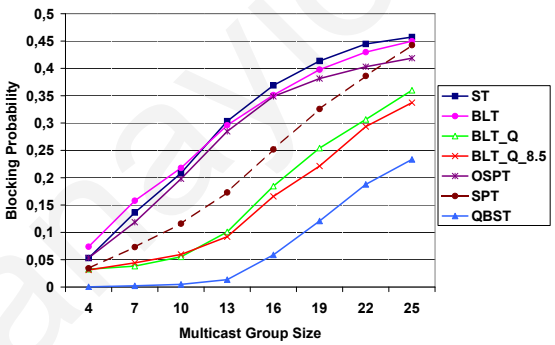

Figure 3.24: Blocking probability versus multicast group size for node engineering with passive splitters.

BLT_$Q_{tolerance}$ and BLT_Q heuristics that also take the PLIs into consideration.

Figs. 3.21 and 3.22 present additional information for the simulation scenario described above by showing the blocking probability due to Q and the blocking probability due to the unavailability of wavelengths, respectively. From these figures, it can be deduced that the QBST heuristic algorithm performs the best as it achieves very low blocking probability due Q. Even though the Steiner tree (ST) heuristic seems to be efficient as far as the recourses of the network are concerned, it does not account for any physical layer constraints, thus resulting in a high blocking probability due to the Q-factor. SPT also does not account for any physical layer constraints but nevertheless has the ability to improve the Q-factor at the tree destinations, as it tends to keep the destinations as close (in distance) to the source as possible. By doing this, SPT tends to decrease the attenuation loss, with the signals also passing through a smaller number of optical amplifiers. OSPT which is an improvement

of the SPT heuristic, succeeds in decreasing the number of links utilized by the tree, but in the expense of increasing the Q-factor at the destination nodes of the tree. This increase of the Q-factor is caused as the algorithm does not consider for the actual distances of the source-destination paths. Finally, the MHT that does not account for the actual distances between the nodes, has an unacceptable high blocking probability, as it tends to increase the attenuation losses of the tree. MDT_F, and DAC heuristics, also have an unacceptable high blocking probability as the target of these two heuristics is to decrease the splitting losses of the tree but without considering the rest of the physical layer impairments. Consequently, both heuristics increase the length of the paths on the tree and in turn the blocking probability due to the Q-factor.

In Fig. 3.23 the results of Fig. 3.20 are repeated for the subset of the heuristics that exhibited reasonable results, while Fig. 3.24 shows the blocking probability versus the multicast group size for node architectures utilizing passive splitters. Results for MHT, DAC and MDT_F heuristic algorithms are omitted from the rest of the analysis since their blocking probability was found to be exceptionally high and since DAC and MDT_F heuristics were developed only for the active splitter case. Examining the results in Figs. 3.23 and 3.24 it is clear that the newly proposed techniques, namely QBST, BLT_$Q_{tolerance}$, and BLT_Q heuristics that take the Q-factor into account perform better for both passive as well as active splitter cases, compared to the rest of the multicast routing heuristics. Again, the QBST heuristic algorithm performs the best for both active and passive cases while the results are slightly better when active splitters are used.

The overall simulation results for these two node architecture scenarios show that there is no particular advantage when using active splitters instead of passive splitters, at least for the engineering scenario that was presented in Section 3.2.1 above. This is due to the fact that VOAs are used to attenuate the total power for each signal to a predetermined value that is calculated based on the worst-case scenario. The results were slightly better (but insignificant better) for active splitters because at the destination nodes the signal power is dropped to the receiver before facing VOA attenuation and this has a slight improvement on the Q-factor. It must be noted here that active splitters are not commercially available as of yet and even when they do become mainstream, their cost and increased complexity (in terms of control) will still tip the scale towards their passive counterparts. As a result, the rest of the performance analysis assumed passive optical splitters at the optical nodes.

3.6.2 Transmitter/Receiver Designs

The following set of figures presents simulation results for the blocking probability versus the multicast group size when a number of multicast tree routing heuristic algorithms are used assuming different node engineering scenarios. The IA-MC-RWA algorithm used for the simulations is now described in Fig. 3.16. In particular, Fig. 3.25 assumes the Case 1 scenario of fixed transmitters/receivers, Fig. 3.26 assumes the Case 2 scenario of the second approach of fixed transmitters/receivers, Fig. 3.27 assumes the Case 3 scenario of tunable transmitters and fixed receivers, Fig. 3.28 assumes the Case 4 scenario of fixed transmitters and tunable receivers, and Fig. 3.29 assumes the Case 5 scenario of tunable transmitters/receivers. The total blocking demonstrated in these cases is a combination of blocking due to the Q-factor, or the unavailability of resources (wavelengths and/or transmitters/receivers). For example, Figs. 3.30 and 3.31 show blocking probability due to Q and blocking probability due to the unavailability of transmitters/receivers respectively, for the Case 1 engineering scenario.

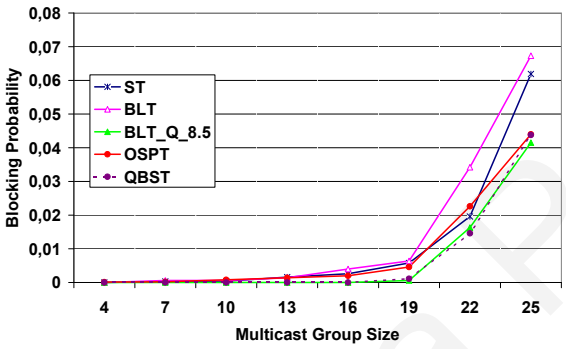

Figure 3.25: Blocking probability versus multicast group size for node engineering with fixed TXs/RXs (Case 1).

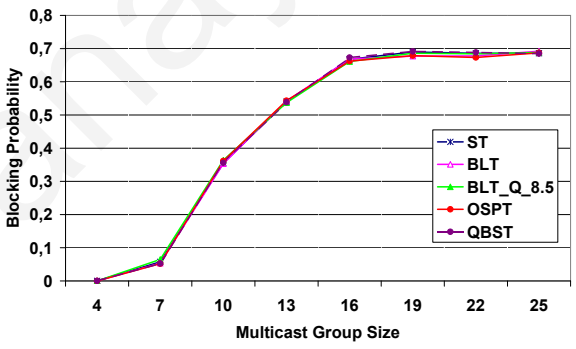

Figure 3.26: Blocking probability versus multicast group size for node engineering with fixed TXs/RXs (Case 2).

Simulation results for the blocking probability versus the multicast group size when a number of multicast routing heuristics are used assuming node engineering with passive splitters for different TXs/RXs designs show that blocking probability is greatly reduced in the fixed TXs/RXs scenario of Case 1, since in this case blocking due to Q is less compared to the rest of the cases (as the worst-case node loss is less in the fixed TX/RX case compared to the cases with tunable components, where switches are also used in the design at the add/drop ports) and there is also more flexibility in the network to assign wavelengths to the multicast connections as there are more TXs/RXs available for wavelength assignment (e.g., in the case of tunable TXs/RXs only 50% of the possible input ports can be dropped at the same time). This is more clearly observed in Fig. 3.31 that shows that the blocking probability

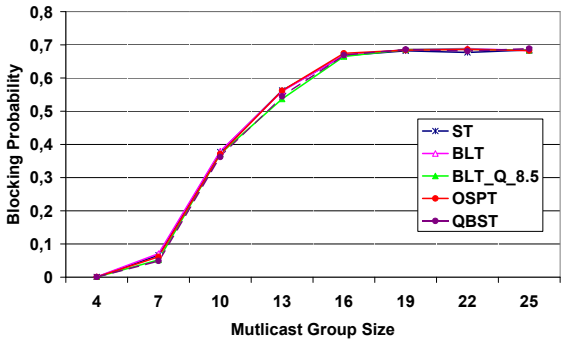

Figure 3.27: Blocking probability versus multicast group size for node engineering with tunable TXs/fixed RXs (Case 3).

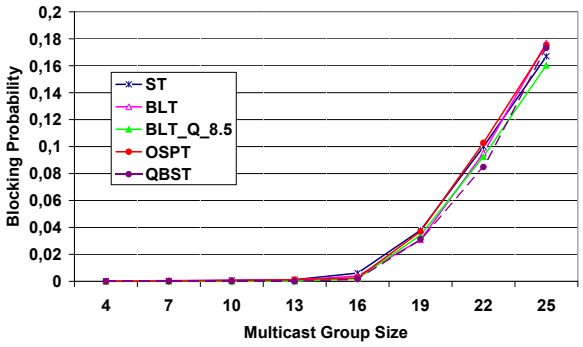

Figure 3.28: Blocking probability versus multicast group size for node engineering with fixed TXs/tunable RXs (Case 4).

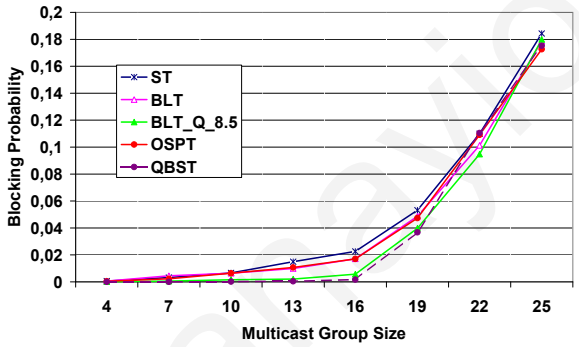

Figure 3.29: Blocking probability versus multicast group size for node engineering with tunable TXs/RXs (Case 5).

due TXs/RXs for every multicast routing heuristic is significantly lower compared to the blocking probability due to the unavailability of TXs/RXs in Cases 2,3,4 and 5 illustrated at Figs. 3.33, 3.35, 3.37 and 3.39 respectively.

The results also show that the QBST and BLT_$Q_{tolerance}$ heuristics that take into account the PLIs perform the best in Cases 1 and 5 since in these cases the blocking probability is not limited by the TXs/RXs constraint and both heuristics manage to limit the blocking probability due to Q-factor. The OSPT is the second best algorithm as it tends to push the destination nodes closer to the source node compared to the other routing heuristics but it also tends to increase the total number of the links in the light-tree (increasing the blocking probability due to unavailability of wavelengths). Figs. 3.30 and 3.38 show the blocking probability due to Q-factor for Cases 1 and 5 respectively, illustrating the advantage of the QBST and BLT_$Q_{tolerance}$ heuristics. In Cases 2-4, all multicast routing heuristics perform almost the same since in these cases the blocking probability is mainly limited by the transmitters/receivers constraint as shown in Figs. 3.32, 3.34 and 3.36 were it is illustrated that

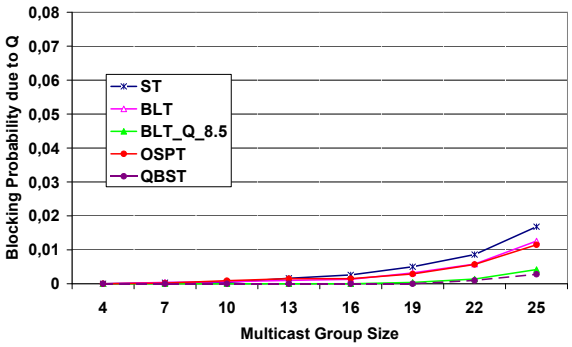

Figure 3.30: Blocking probability due to Q versus multicast group size for node engineering with fixed TXs/RXs (Case 1).

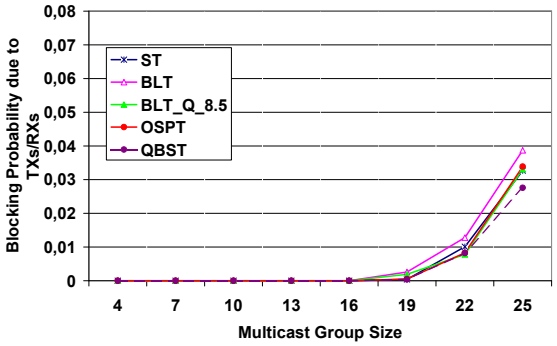

Figure 3.31: Blocking probability due to the unavailability of TXs/RXs versus multicast group size for node engineering with fixed TXs/RXs (Case 1).

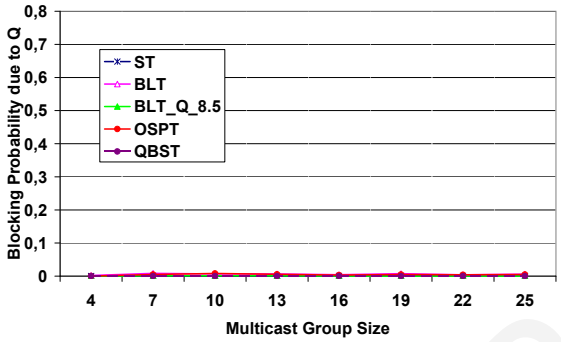

Figure 3.32: Blocking probability due to Q versus multicast group size for node engineering with fixed TXs/RXs (Case 2).

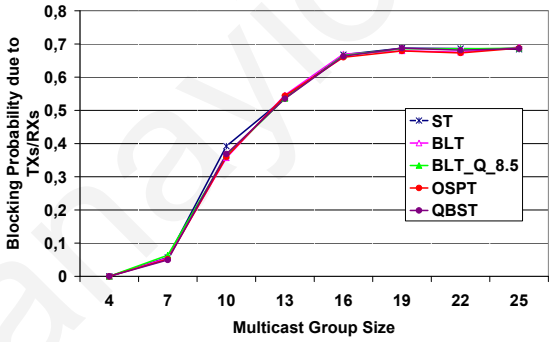

Figure 3.33: Blocking probability due to the unavailability of TXs/RXs versus multicast group size for node engineering with fixed TXs/RXs (Case 2).

the blocking probability due to Q is insignificant compared to their corresponding blocking probability due to the unavailability of TXs/RXs.

The highest blocking probability among all cases is exhibited in Cases 2 (fixed transmitters/receivers but with some limitations on their numbers) and 3 (tunable transmitters/fixed receivers). For these cases there is also no difference in terms of performance when different routing techniques are utilized. This is the case as for these two cases blocking is mainly due to the lack of resources rather than due to Q, thus the proposed routing techniques that have an advantage in terms of physical layer impairments (PLIs) are performing similar to the ones that do not take the PLIs into consideration. Furthermore, comparing the cases where tunable transmitters/fixed receivers and fixed transmitters/tunable receivers are used respectively one can deduce that it is more important to have more available receivers than transmitters for multicast connectivity, as the blocking probability for the latter case was

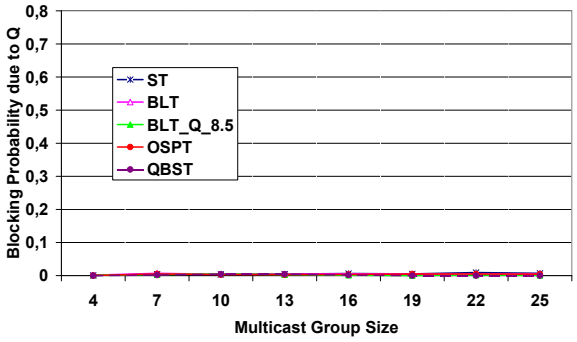

Figure 3.34: Blocking probability due to Q versus multicast group size for node engineering with tunable TXs/fixed RXs (Case 3).

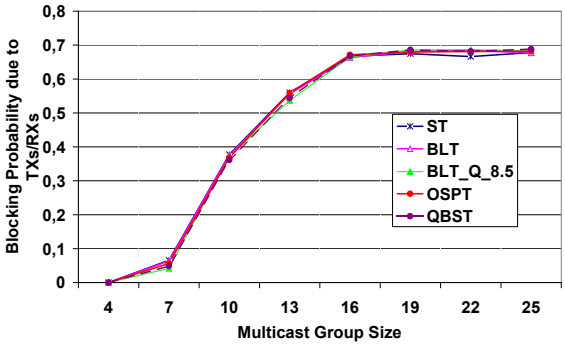

Figure 3.35: Blocking probability due to the unavailability of TXs/RXs versus multicast group size for node engineering with tunable TXs/fixed RXs (Case 3).

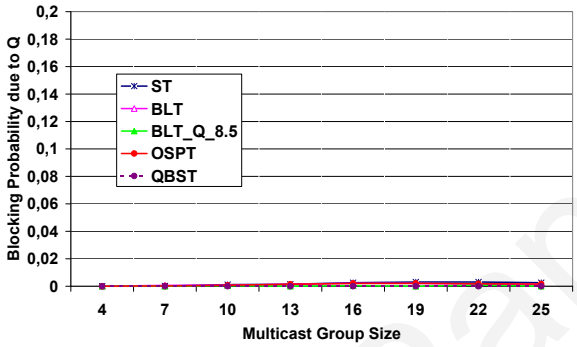

Figure 3.36: Blocking probability due to Q versus multicast group size for node engineering with fixed TXs/tunable RXs (Case 4).

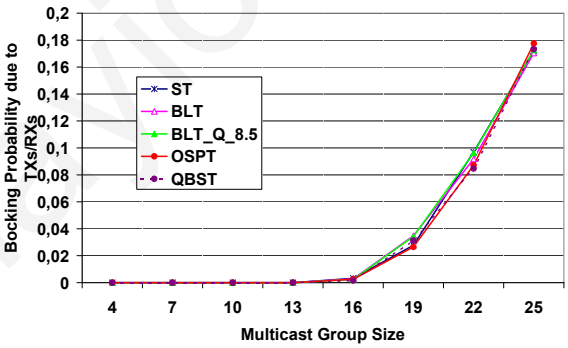

Figure 3.37: Blocking probability due to the unavailability of TXs/RXs versus multicast group size for node engineering with fixed TXs/tunable RXs (Case 4).

significantly lower compared to the former.

The performance of the D-IA-MC-RWA algorithm is also evaluated for the two TXs/RXs designs that exhibited the best results, namely the fixed TXs/RXs design of Case 1 and the tunable TXs/RXs design of Case 5. The performance of the D-IA-MC-RWA algorithm was evaluated for several multicast routing heuristics, and compared to the IA-MC-RWA that also accounts for the TXs/RXs constraints (Fig. 3.16).

These results are shown in Figs. 3.40 and 3.41. Fig. 3.40 shows that the ST heuristic performs slightly better compared to the other multicast routing heuristics for large group sizes, while Fig. 3.41 shows that all multicast routing heuristics perform almost the same as in this case the blocking probability is limited by the number of TXs/RXs. Note that in Fig. 3.40 the performance of the different multicast routing heuristics is exactly the opposite compared to the case were the conventional IA-MC-RWA is utilized. For example, BLT_$Q_{tolerance}$

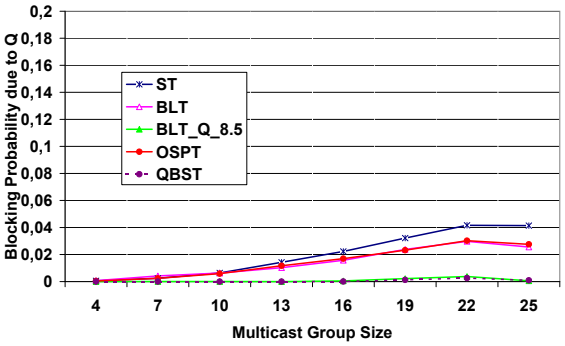

Figure 3.38: Blocking probability due to Q versus multicast group size for node engineering with tunable TXs/RXs (Case 5).

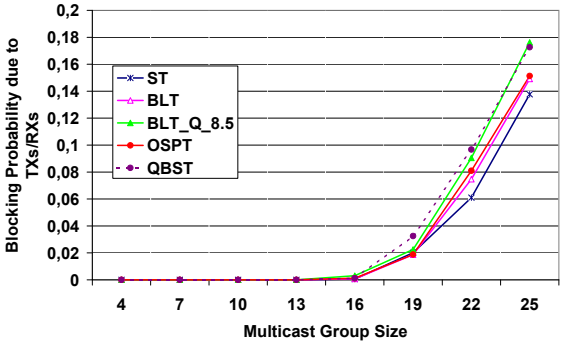

Figure 3.39: Blocking probability due to the unavailability of TXs/RXs versus multicast group size for node engineering with tunable TXs/RXs (Case 5).

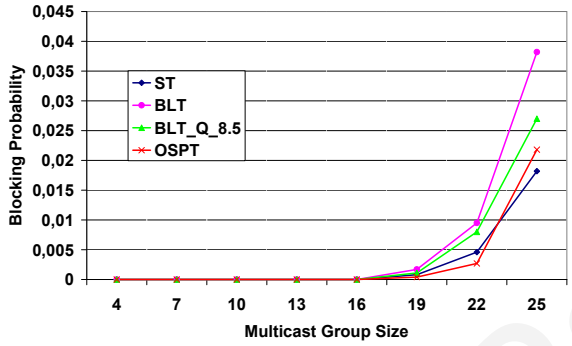

Figure 3.40: Blocking probability versus multicast group size with fixed TXs/RXs (Case 1) (D-IA-MC-RWA heuristic algorithm is utilized).

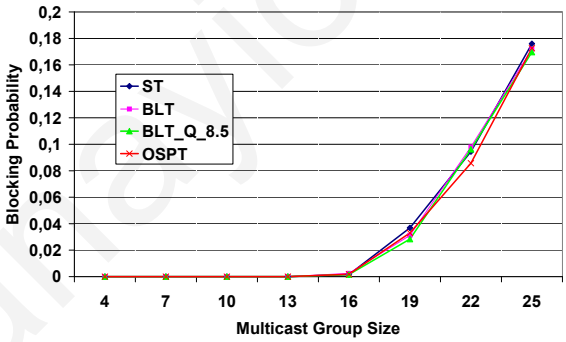

Figure 3.41: Blocking probability versus multicast group size with tunable TXs/RXs (case 5) (D-IA-MC-RWA heuristic algorithm is utilized).

performs better than ST in the conventional IA-MC-RWA algorithm, while in the D-IA-MC-RWA, ST performs better than BLT_$Q_{tolerance}$. This is due to the fact that BLT_$Q_{tolerance}$ aims at reducing the average Q-factor of the multicast tree but increases the number of links in the tree. The D-IA-MC-RWA, however, improves the Q-factor for every multicast routing heuristic and thus the ST heuristic algorithm that returns trees of minimum cost, performs the best.

As pointed out, the D-IA-MC-RWA algorithm tends to increase the number of TXs utilized by a single multicast call. The question that arises here is if the D-IA-MC-RWA algorithm increases the blocking probability due to the unavailability of TXs, to the point that the algorithm becomes inefficient compared to the conventional IA-MC-RWA algorithm that assigns only one TX per multicast call. Figs. 3.42 and 3.43 show the blocking probability versus the multicast group size for Cases 1 and 5 respectively. In each figure, the ST

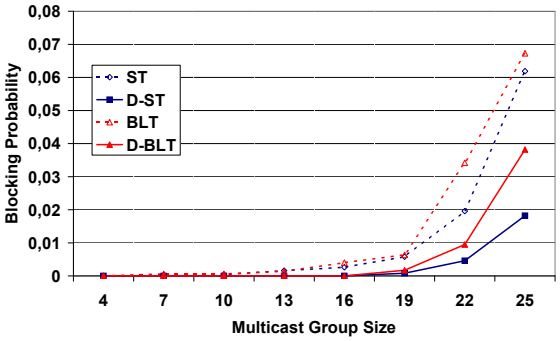

Figure 3.42: Blocking probability versus multicast group size with fixed TXs/RXs (Case 1) for IA-MC-RWA and D-IA-MC-RWA heuristic algorithms.

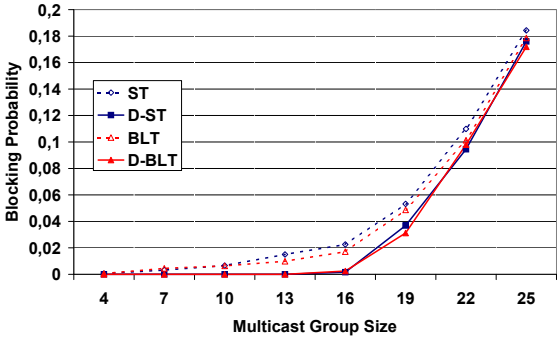

Figure 3.43: Blocking probability versus multicast group size with tunable TXs/RXs (Case 5) for IA-MC-RWA and D-IA-MC-RWA heuristic algorithms.

and BLT heuristics are compared for both the conventional IA-MC-RWA algorithm and the D-IA-MC-RWA algorithm. Note that when IA-MC-RWA is assumed, the multicast routing heuristics are denoted by ST and BLT, while when the D-IA-MC-RWA algorithm is assumed the heuristics are denoted by D-ST and D-BLT. Results show that D-IA-MC-RWA algorithm outperforms the conventional IA-MC-RWA for both multicast routing heuristics, and although not shown in Figs. 3.42 and 3.43, results previously given in this chapter for the IA-MC-RWA algorithm demonstrate that the D-IA-MC-RWA performs better for every multicast routing heuristic examined here. Considering again Figs. 3.42 and 3.43 it can be deduced that the gain of utilizing the D-IA-MC-RWA algorithm for the fixed TXs/RXs design (Case 1) is greater compared to the tunable TXs/RXs case, as in the former case more transmitters are available in the network.

Even though it seems that the decomposed approach achieves better results than the cases where the entire light-tree is provisioned on a single wavelength utilizing a single TX, it must be noted that the decomposed approach has an additional cost and complexity associated with it in terms of the use of additional electronic equipment as well as controlling these equipment to be able to send the same information utilizing more than one TX on separate wavelengths, by creating several sub-light-trees.

3.6.3 PDG/PDL Performance Results

Since as was shown above designs corresponding to Cases 1 and 5 perform better compared to the other node architectures, the rest of the performance analysis, additionally accounting for the PDG/PDL effect, assumes only these two cases.

Figs. 3.44 and 3.45 show the blocking probability versus the multicast group size for the case of fixed transmitters/receivers and cover two of the three case studies previously described, namely the best-case scenario and the worst-case scenario. In the best-case scenario the signal/component polarizations are aligned to the maximum gain/minimum loss polarization axes while in the worst-case scenario they are aligned to the minimum gain/maximum loss polarization axes. In the best-case scenario, the performance of the four routing heuristic algorithms (ST, BLT, BLT_Q$_{tolerance}$, and QBST) is slightly improved compared to the case where PDG/PDL was not considered (Fig. 3.25), and this is due to the unpolarized ASE noise which experiences less net gain than the signal in the best-case scenario (becoming partially polarized), and thus results in slightly higher OSNR and less blocked calls. In other words, in the best-case scenario the signal/component polarizations are aligned to the maximum gain/minimum loss polarization axes and therefore the signal experiences the same gain/loss as in the case where the PDG/PDL effect was not considered. What results in the improved blocking is the noise component that is unpolarized. Unlike the no PDG/PLD case where the noise was assumed to be affected by the same gain/loss as the signal, in the best-case scenario the ASE noise is averaged over the two polarization axes of the model in Figs. 3.10 and 3.11 resulting in a noise component that is better compared to the no PDG/PDL case. Hence, the overall Q-factor is improved in the best-case. This effect was also observed in [9].

Also, in best-case scenario, the QBST heuristic algorithm performs the best, with the second best algorithm being the ST heuristic that outperforms the BLT_Q$_{tolerance}$ heuristic that accounts for the PLIs. The reason for this is simply the fact that in the best-case scenario the physical layer effect is insignificant and therefore the ST heuristic that builds minimum cost trees performs better than the BLT_Q$_{tolerance}$ technique that tends to build trees with increased cost but lower Q-factor.

For the worst-case scenario the performance is a lot worse for all four heuristics compared to the best- and no PDG/PDL-cases. The limiting factor in this case is the Q-factor and therefore the QBST and BLT_Q$_{tolerance}$ heuristics are performing significantly better than the rest of the multicast routing heuristics. Also, QBST outperforms the BLT_Q$_{tolerance}$ heuristic as it manages, more effectively, to decrease the Q-factor at the destination nodes.

Fig. 3.46 shows the third case study, namely the random case, where the polarization axes are randomly varied as discussed before. The performance is significantly improved compared to that of the worst-case study with BLT_Q$_{tolerance}$ outperforming the rest. Although in best- and worst-case scenarios the QBST heuristic was outperforming the rest of the multicast routing heuristics, now the BLT_Q$_{tolerance}$ approach is the best and this is due to the fact

that the blocking probability here is affected by both the Q-factor and the cost of the tree. This is more clearly illustrated in Fig. 3.48, where it is shown that the impact of the Q-factor gives an advantage to both QBST and BLT_$Q_{tolerance}$ heuristic algorithms, while Fig. 3.49 shows that the impact of the number of links used for the tree gives an advantage to the BLT_$Q_{tolerance}$ heuristic. Fig. 3.47 shows that no significant result variability is obtained for different simulation samples of randomized polarizations for group sizes up to 19. However, for larger group sizes the blocking probability slightly improves as the sample size increases, in which case the polarization scenario becomes more realistic.

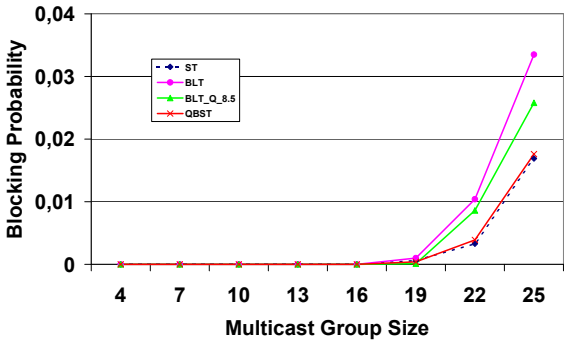

Figure 3.44: Blocking probability versus multicast group size for the best-case PDG/PDL scenario and node engineering with fixed TXs/RXs (Case 1).

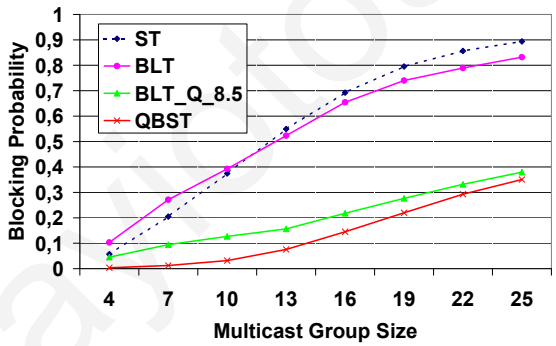

Figure 3.45: Blocking probability versus multicast group size for the worst-case PDG/PDL scenario and node engineering with fixed TXs/RXs (Case 1).

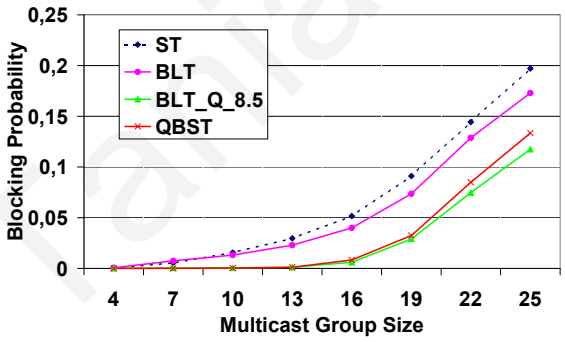

Figure 3.46: Blocking probability versus multicast group size for the random-case PDG/PDL scenario and node engineering with fixed TXs/RXs (Case 1).

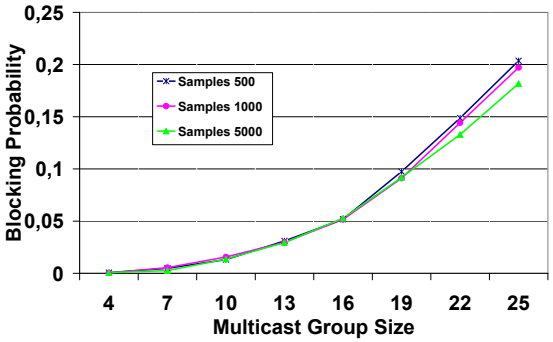

Figure 3.47: Blocking probability versus multicast group size for the random-case PDG/PDL scenario and node engineering with fixed TXs/RXs (Case 1). Only the results of the ST heuristic algorithm are shown for different simulation samples of randomized polarizations.

For a typical multicast group size of 22 and considering the best QoT multicast routing

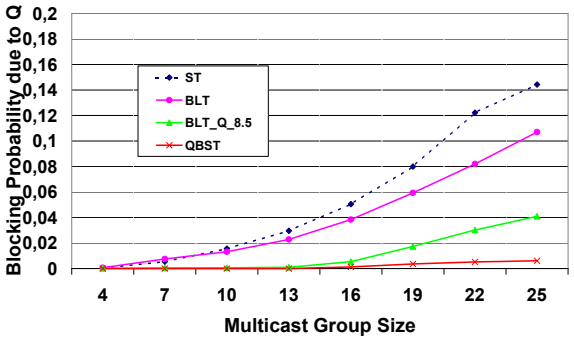

Figure 3.48: Blocking probability due to Q versus multicast group size for the random-case PDG/PDL scenario and node engineering with fixed TXs/RXs (Case 1).

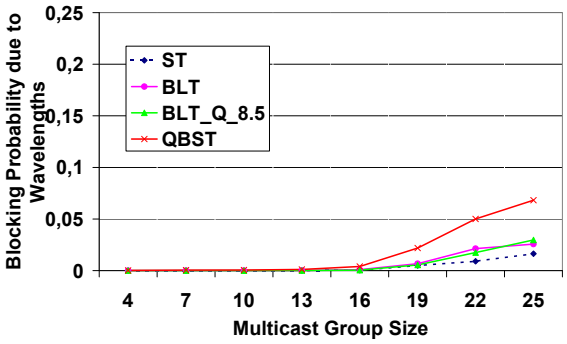

Figure 3.49: Blocking probability due to the unavailability of wavelengths versus multicast group size for the random-case PDG/PDL scenario and node engineering with fixed TXs/RXs (Case 1).

heuristic (BLT_$Q_{tolerance}$ in this case), a large variability in the resulting blocking probability is observed in Fig. 3.50. The best-case, which is very close to the no PDG/PDL results exhibits less than 1% blocking whereas the worst-case has an unacceptable 33% blocking. Note that the no PDG/PDL case, which is the most optimistic scenario and is illustrated in Figs. 3.50 and 3.51 refer to the case where both signal and noise were always aligned to the minimum loss/maximum gain at each component in the network, and thus the effect of PDG/PDL was not considered. The no PDG/PDL results presented here correspond to the results presented in Section 3.6.2. The more realistic random polarization scenario yields a blocking of about 7% which is still a significant variation from the no PDG/PDL results. Fig. 3.51 shows the results (again for BLT_$Q_{tolerance}$) for the case of a different node architecture which now includes tunable TXs/RXs (Fig. 3.9) in the network. The available number of TXs/RXs is now equal to the number of wavelengths in the system, N. The results again demonstrate a large blocking variability among the cases and a higher overall blocking which is mostly due to the unavailability of TXs/RXs. It must be noted that in both the scenarios of Figs. 3.50 and 3.51 the VOAs could not handle PDG/PDL due to their inability to distinguish ASE noise from signal.

This work has basically shown that the effect of PDL/PDG is extremely important in the provisioning calculations and cannot be ignored or budgeted in a simple manner when provisioning all-optical multicasting connections in a network. For typical PDL/PDG component values a large performance variation will be introduced. Provisioning for a worst-case scenario, which might be the solution in this case will produce unacceptable results in terms of blocking probability. However, the worst-case approach is not always an obvious choice.

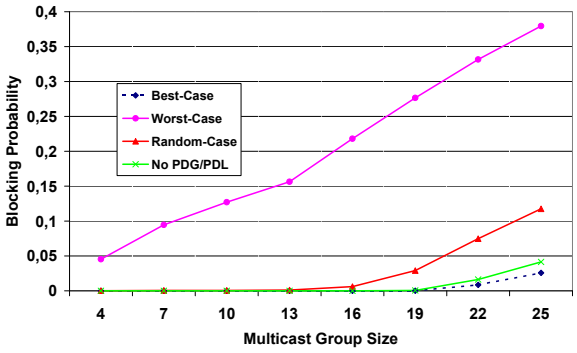

Figure 3.50: Multicasting results using BLT_$Q_{tolerance}$ algorithm with fixed TXs/RXs (Case 1).

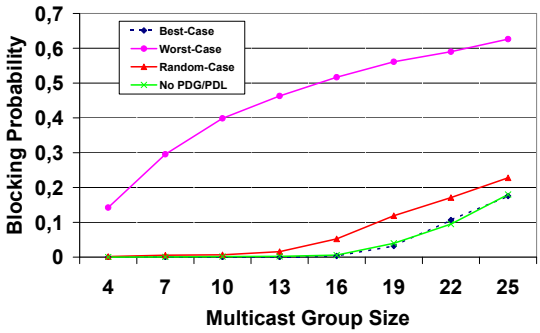

Figure 3.51: Multicasting results using BLT_$Q_{tolerance}$ algorithm with tunable TXs/RXs (Case 5).

PMD was shown to be treated using a maxwellian distribution approach in system simulations in the past. The above has an inherent averaging approach and is not a worst-case scenario. As a result, looking at the average and best-case scenarios and obtaining the variation in blocking probabilities that result utilizing the different routing heuristics, is a very interesting comparison. Furthermore, this is the first time that the above effect has been introduced to the study of management and transport layer interactions in transparent optical networks.

3.7 Conclusions

In this chapter the impact of node design/engineering, as well as the impact of the physical layer constraints, via the Q-factor, on the multicast routing heuristics is shown. It is demonstrated via simulation that different node designs/engineering at the physical layer produces different multicast group blocking, a strong indicator that a more refined interaction between physical and logical layer is needed for multicast connection provisioning. Specifically, from the performance results it is clear that a passive splitters design will not affect the network performance (compared to an active splitters design where the node has control on the splitting of the signal) and that more flexibility in terms of transmitters and receivers significantly decreases the blocking probability. Also, the choice of the routing heuristic greatly impacts the network performance, with the newly proposed QBST and BLT_$Q_{tolerance}$ heuristic algorithms outperforming all routing approaches that do not consider the physical layer effects, as well as the routing heuristics that only account for the power budget.

Furthermore, the introduction of PDG/PDL in the optical multicast routing heuristics

using physical layer constraints has demonstrated an increased complexity that renders the conventional probabilistic handling of PDG/PDL not practical. The randomized polarization approach which is the most realistic in such a system shows significant rise in the blocking probability of multicast connections compared to the studies where no PDG/PDL effects were included. Clearly, in this case another layer of interaction is needed and it involves dynamic gain equalizers as outlined in Chapter 7 of this thesis.

Chapter 4

Protection of Multicast Sessions in Transparent Optical Networks with Mesh Topologies

In wavelength routed networks, apart from the efficient provisioning of multicast sessions, as demonstrated in Chapter 3, it is also important to maintain the survivability of the sessions in the presence of faults. As fiber-optic communication systems are cable-based systems, the most prevalent form of failure is the failure of a fiber-optic link due to a variety of reasons (backhoe effect, human error, natural effects, etc). When a fiber-optic link fails, all the information carried by that link is lost (each fiber-optic link carries a large number of wavelengths with a very high aggregate capacity). As multicast sessions carry traffic to multiple destinations, the impact of a link failure is even more severe compared to the impact of a link failure on a unicast session. Thus, it is even more critical to protect multicast sessions against single link failures.

The problem of providing survivability on unicast connections has been extensively studied in the literature [18, 24–26, 53, 57, 58, 65, 66, 74, 75, 79, 106, 111, 118, 146, 147, 180, 202, 220], and the myriad of approaches investigated will not be reiterated here. As in this chapter the problem of protecting multicast sessions in transparent optical networks is examined, the state-of-the-art overview will focus on existing multicast protection schemes. This will be followed by newly proposed multicast protection techniques for segment and cycle-based approaches. The performance of these multicast protection schemes will be examined when both the PLIs are not considered and when the PLIs are taken into account. Multicast protection techniques with physical layer impairments are investigated for the first time in this

dissertation providing a very useful guideline on which general techniques should be considered in real network applications and what should be the characteristics of efficient protection techniques in transparent optical networks.

4.1 Multicast Protection State-of-the-art

The key objective of multicast traffic protection is to ensure that after the failure every affected destination can receive the information from the source via the protection path(s) quickly and without requiring a large amount of redundant capacity that is reserved in the network and utilized only when a failure occurs. In general, failure recovery in optical networks falls under two categories, namely *protection* where the recovery paths are precomputed prior to the failure occurrence, and *restoration* where the recovery paths are computed dynamically after the failure has occurred. In this thesis only protection techniques are considered (the most prevalent form of recovery in mesh optical networks mainly due to the speed of recovery compared to the restoration approaches).

Furthermore, several (general) protection schemes exist for protecting the multicast traffic, and these schemes fall mainly into three categories namely link-based, path-based, and segment-based. In link-based protection approaches (cycle-based approaches also fall under this category) the recovery path is found between the two endpoints of the failure, in path-based protection approaches a disjoint recovery path is found between the source and destination nodes, and in segment-based protection techniques (which fall in-between the link- and path-based approaches) part of the primary path is used as the recovery path and a backup path is found for a segment of the primary path (the smallest segment is the failed link as is the case of link-based protection, whereas the largest segment is the entire path from source to destination, as is the case of path-based protection). Protection techniques can also be categorized in terms of their redundant capacity sharing approaches, resulting in the *dedicated* and *cross-shared* protection schemes [172].

In dedicated backup protection (i.e., one-plus-one (1+1), one-for-one (1:1) [24]) the resources along the backup path(s) are dedicated for only one connection and are not shared with the backup paths for other connections. In case of path-based dedicated (1+1) protection the traffic is bridged at the transmitter to both the working and backup path(s), and one of the two signals is selected at the receiver(s) using a switch. If a failure occurs, the switch at the receiver end is used to switch to the signal coming from the secondary path, thus very quickly recovering from the failure. One-for-one protection (1:1) is similar to the

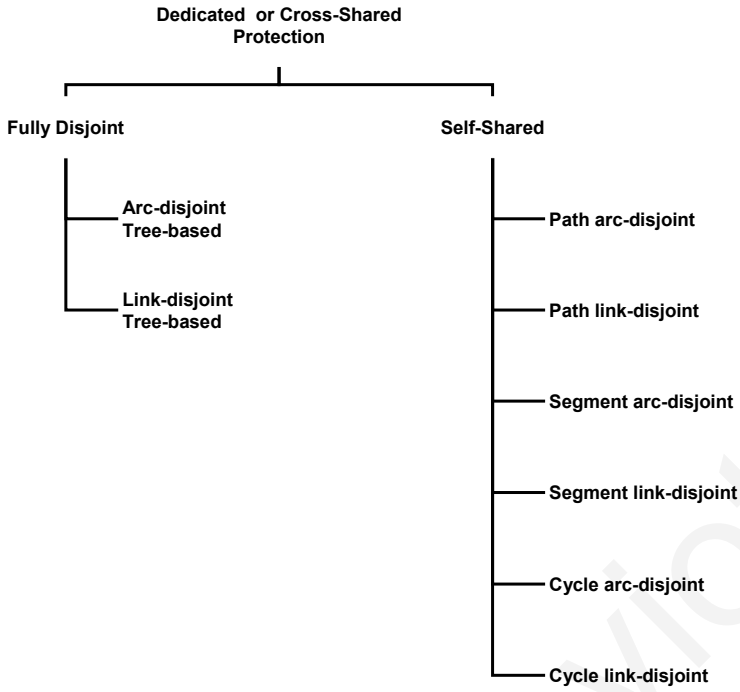

Figure 4.1: Classification of protection techniques in terms of their sharing schemes.

(1+1) dedicated protection with the difference that in this case the traffic is not bridged at the source node but is switched either at the working or backup path(s). If a failure occurs, both transmitter and receiver ends switch to the secondary path(s) to recover from the failure. Although dedicated approaches provide a very fast recovery service are not capacity efficient since the redundant capacity required will in general exceed 100 percent [24].

In cross-shared backup protection schemes [172], resources along a backup path may be shared with other backup paths. The constraint in these schemes is that the different backup path(s) can share the common backup wavelengths if their corresponding working path(s) are disjoint. As a result, backup channels are multiplexed among different failure scenarios, which are not expected to occur simultaneously (the assumption in almost all the failure recovery schemes are for single failure scenarios as the probability of more than one failures occurring simultaneously in the network is extremely small), and therefore, this approach is more capacity efficient but slower compared to dedicated backup protection schemes.

As shown in Fig. 4.1, in both dedicated and cross-shared protection schemes, either fully *disjoint* or *self-sharing* techniques can be applied depending on the multicast protection approach that is followed. *Disjoint* techniques refer to the case where the backup path(s) of a multicast session are fully link- or arc-disjoint from the working path(s) of the multicast

session. *Self-Sharing* technique [172] refers to the case where the backup paths can share the working wavelengths on the working tree of the same multicast session, thus improving the network resource utilization. There are two types of self-sharing. When a backup path shares bandwidth or channels on common edges with other backup paths on the backup tree, this is called *intra-tree* sharing (something that is done by default on the primary tree). A backup path can share edges not only with other backup paths, but also with other edges on the primary tree. When there is sharing of edges between the primary and the backup tree, this is called *inter-tree* sharing.

In the sections that follow various existing multicast protection techniques are presented and analyzed and these are grouped into tree-based, path-based, segment-based, and cycle-based techniques.

4.1.1 Tree-based Protection Techniques

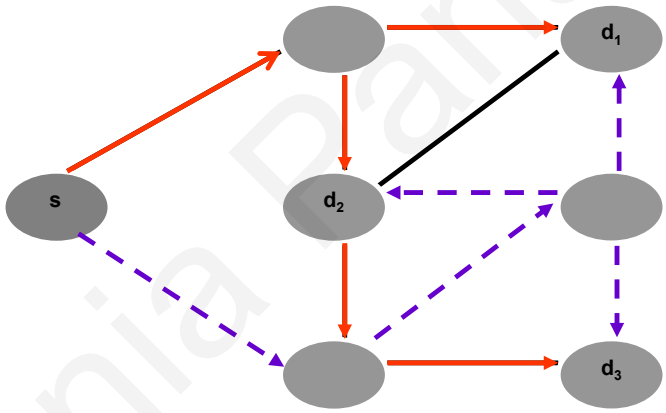

Figure 4.2: Arc-disjoint trees.

A straightforward approach for protecting a multicast tree is the tree-based approach in which two link-disjoint backup trees, that do not share any link along their edges, are computed. Pitfalls of this approach include excessive use of resources and inability to discover link-disjoint working and backup light-trees in a mesh network, which may lead to a large number of blocked multicast sessions [173]. The usage of network resources can be improved if two arc-disjoint trees are computed, that unlike link-disjoint trees utilize only one of the two directions of the link. Thus a link can be shared in opposite directions only (Fig. 4.2). Arc-disjoint trees, however, do not yield the optimal solution for efficiently pro-

tecting a multicast session, and the connection may be blocked.

In general, finding the working and the backup trees corresponds to finding two Link-Disjoint Trees (LDT) or Arc-Disjoint Trees (ADT). The basic steps of LDT and ADT protection algorithms are shown in Tables 4.1 and 4.2 respectively.

Table 4.1: LDT Protection Algorithm

Step 1	Compute a working tree T from network graph G.
Step 2	Remove all links along the working tree to create graph G'.
Step 3	Compute a backup tree T' in graph G'.

Table 4.2: ADT Protection Algorithm

Step 1	Compute a directed working tree T from network graph G.
Step 2	Remove all arcs along the working tree to create graph G'.
Step 3	Compute a backup tree T' in graph G'.

For the computation of working and backup trees authors in [171] used the Pruned Prim Heuristic (PPH) [44] and the Minimum-cost Path Heuristic (MPH) [187] for both the link-disjoint and arc-disjoint cases, while authors in [141] used a minimum cost heuristic (MCH) algorithm that is based on Dijkstra's algorithm for the link-disjoint case to provide 1+1 protection. In [141] the MC-APE algorithm was also proposed to provide 1+1 protection, aiming at reducing the chance of creating an isolated node and consequently increasing the blocking probability for multicast sessions, by randomly removing an edge (link) from the source node and from all destination nodes before the computation of the working tree. The authors in [88, 89] proposed the Optimized Shortest Paths (OSPT) heuristic, for the computation of both working and backup link-disjoint multicast trees, providing 1+1 dedicated protection. The OSPT technique aims to reduce the size of the multicast tree and maximize the bandwidth reuse without increasing the computational complexity for finding the tree. OSPT is based on Dijkstra's shortest path algorithm to find the minimum-cost path between the source and each multicast destination node. The cost of all links included in an already-found path is set to zero in order to maximize capacity reuse. In the end, all minimum-cost paths are merged together to construct the multicast tree.

4.1.2 Path-based Protection

In path-based schemes, for each destination node, a backup path, which is link-disjoint or arc-disjoint from the working path from the source to that destination node in the multicast

tree, is derived. There are two main approaches to path protection. In the more basic approach the primary tree is computed first and then backup paths from the source to each destination node that are link- or arc-disjoint from the primary path to that destination in the primary tree are computed. One drawback of this scheme is that only those paths that originated at the source node of the multicast tree and ended at the destination nodes are considered, thus this technique cannot derive the most efficient backup paths. Another approach to provide preplanned path protection is to compute path-pairs from a source node to all destination nodes of a multicast session. In this case, the primary and backup paths are computed together, one (source)-destination pair at a time. This approach is called path-pair protection because it assures that each destination can be reached by a path-pair of primary and backup paths [174]. The set of all primary paths forms the primary multicast tree, while the set of all backup paths forms the backup tree. The path-pairs in the tree can be link-disjoint or arc-disjoint. The key aspect of this approach is the fact that when a new source-destination primary path is computed it can reuse existing links and channels/bandwidth from other existing source-destination primary paths. A distinction with LDT- and ADT-based protection is that, with path-based approaches, a backup path can follow the same links/arcs that are part of the primary tree created so far, as long as it is link/arc-disjoint for that particular destination. The disadvantage of this approach is that in general it uses more links to set up primary and backup trees compared to the arc-disjoint and the link-disjoint tree-based protection schemes previously described. Table 4.3 describes the basic steps of the path-pair protection algorithm. Note that several heuristics exist for the ordering scheme of Step 1.

Table 4.3: Path-pair Protection Algorithm

Step 1	Place all source-destination demands in a list following a specific ordering scheme.
Step 2	Following the order, for every source-destination pair repeat Steps 3 and 4.
Step 3	Find a least-cost path-pair (link- or arc-disjoint) between the source and destination node in network graph G.
Step 4	Reset the cost of the links to 0 for links on primary paths for already found path-pairs.

Various path-based protection schemes have been proposed in [88, 96, 108, 141, 170, 174, 200]. Authors in [170] proposed a path protection algorithm called Improved Path (Table 4.4). According to this scheme, each primary path in the tree is protected by an arc-disjoint backup path. The algorithm introduces cross- and self-sharing by setting to 0 the cost of the links on the primary tree and of the already-found backup paths. Therefore, the

advantage of the Improved Path approach comes from the increased sharing possibility of the backup path.

Table 4.4: Improved Path Heuristic [170]

Step 1	Create a primary tree T using the minimum cost path heuristic.
Step 2	Set the cost of all edges on the primary tree T to 0.
Step 3	For every destination node of the session, repeat Steps 4 and 5.
Step 4	Compute a backup path disjoint from the primary path from source to the destination node.
Step 5	Update the cost for links on already-found backup path to 0.

In [174], a path-pair protection scheme is proposed called Optimal Path Pair-based Shared Disjoint Paths (OPP-SDP). This scheme uses the general pattern of path-pair protection as described in Table 4.5. The OPP-SDP heuristic always finds the least-cost path-pair between two endpoints if such a pair exists. This algorithm is derived from Suurballe's algorithm [186]. Once a path-pair is found, the cost of the arcs along it are updated to zero to increase sharing of new path-pairs with the already-found ones and to minimize additional cost.

Table 4.5: OPP-SDP Heuristic [174]

Step 1	For every destination node repeat Steps 2 and 3.
Step 2	Find optimal path-pair between source and destination nodes.
Step 3	Reset cost for already found optimal path-pairs to 0.

The work in [88], which is an extension of the algorithms described in [141], proposes an arc-disjoint path protection algorithm called Optimized Collapsed Rings (OCR). The OCR scheme is significant because it does not compute a backup tree but traverses the primary tree backwards instead. The primary tree is computed by starting from the source node and visiting all destinations in an optimized order. The first destination visited after the source is the one closest to it. The algorithm computes the shortest path between the source and the destination. All links along the path are removed and the destination becomes the new source. After that, the path is expanded to include the second closest destination and so on until all destination nodes are included in the primary tree. For the backup path, the OCR algorithm computes the shortest path between the original source and the last destination visited and then traverses the primary tree backwards until the first destination is reached, to create a backup tree. This scheme is extremely efficient in terms of resource utilization since the primary and the backup trees are almost identical.

4.1.3 Segment-based Protection

In segment-based protection schemes the working tree is first divided into segments using different techniques that are specified in the protection scheme, and then each segment is protected separately by any applicable protection technique. The most critical problem for this approach relies on how the segments of a multicast tree are identified, as different identification techniques result to different network performance [107].

A segment in a primary light-tree can be defined as the path between two segment points of the tree and segment points can be defined, for example, as all the splitting nodes, the destination nodes, and the source node. Once the segments points and segments are identified, each segment of the primary tree is protected by discovering a backup segment that is link- or arc-disjoint from its corresponding primary segment [170]. Table 4.6 describes the basic steps of a segment-based protection algorithm. Various segment-based protection algorithms have been proposed in [61, 104, 105, 107, 170, 174, 198, 236].

Table 4.6: Segment-Based Protection Algorithm

Step 1	Compute a primary tree T from network graph G.
Step 2	Identify the primary segments on the primary tree T.
Step 3	For every primary segment repeat Step 4.
Step 4	Compute a backup path that is disjoint from the primary segment using a shortest paths heuristic (e.g., Dijkstra's algorithm).

The authors in [174] proposed the Shared Disjoint Segments (SDS) protection algorithm. In that case, a segment was defined as a sequence of arcs between the segment points of the tree, while segment points were defined as the source node, the splitting nodes, and the destination nodes of the primary tree. The SDS scheme expands on the general segment protection algorithm described in Table 4.6 by computing arc-disjoint backup segments that use both intra- and inter-tree sharing techniques. In Step 4 of the algorithm the SDS scheme implements ADT protection as described in Table 4.2 for each backup segment. When the algorithm tries to find a backup segment for an unprotected segment on the primary tree, the cost of the arcs along the primary tree and already-found backup segments is updated to zero, thus enhancing sharing of the current backup segment with the partially computed subgraph. As a result, the additional cost for computing each new backup segment is minimized.

A modification of the SDS heuristic described in Table 4.7 is proposed in [170], namely the Improved Segment heuristic. The Improved Segment heuristic, instead of computing a single backup segment for each primary segment, it computes multiple backup segments for each primary segment and selects the best backup segment available. This increases

Table 4.7: SDS Heuristic [174]

Step 1	Compute a primary tree T from network graph G.
Step 2	Identify the primary segments on the primary tree T.
Step 3	Make cost= 0 for the arcs along the primary tree.
Step 4	For every primary segment repeat Steps 5 through 8.
Step 5	Remove the links along the primary segment.
Step 6	Compute a backup segment arc-disjoint to the primary segment.
Step 7	Update cost= 0 for already found backup segments.
Step 8	Replace the links along the primary segment.

the complexity of the algorithm but provides better resource utilization because it computes multiple backup segments for each primary segment and selects the best backup segment available. More precisely, it computes a least-cost backup segment from any upstream node to the downstream segment end-node and selects the backup segment with the least-cost amongst them.

Another segment-based heuristic algorithm was also proposed in [105], called Adaptive Shared Segment Protection (ASSP), which determines the segments during the recovery process. The key idea behind the ASSP heuristic is that a tree does not contain any cycles and that a multicast tree must contain at least two destination nodes. The authors claim that if a path is computed between two destination nodes that are related by the same splitting node (common ancestor), the segments formed by both destination nodes and their common splitting node will be protected via the cycle that is formed between the segments and the backup path. Therefore, the ASSP algorithm first builds the primary multicast tree and then creates a set of all destination nodes. Then it tries to find a shortest path between any two destination nodes that is link-disjoint from the primary multicast tree. The cost of the links in an already-found shortest path is updated to zero in order to maximize intra-tree sharing. The algorithm then selects as backup paths the shortest paths that protect the maximum possible number of relevant segments with the least possible amount of resources.

Authors in [107] proposed another segment-based protection scheme called segment-based protection with sister node first (SSNF). Its basic idea is to protect a primary light-tree using a set of backup segments, with higher priority to protect the segments from a splitting node to its children. In this scheme, sister nodes are defined as the nodes with a common parent and segment points are defined as the splitting nodes or the nodes with degree 1. Initially, a primary light tree T is created and an auxiliary graph AG is created consisting of the segment points and the segments of the primary tree T. Once the sister nodes are identified on AG, the SSNF heuristic tries to protect the segments from each branch point to

its children using a least-cost tree. If such a least-cost tree does not exist, then the segment protection approach is used (e.g., Step 2 to Step 4 of Table 4.7) to protect each segment individually. The SSNF algorithm includes both inter- and intra-tree sharing because the cost of all links, which are either part of the primary tree or are already used as backup links, is set to 0.

4.1.4 Cycle-based Protection

Ring-based protection techniques originated from SONET rings that defined two types of self-healing rings (SHR), namely the bi-directional line switched ring (BLSR) and the uni-directional path-switched ring (UPSR). Two types of BLSR architectures exist, namely the 2-fiber (2F-BLSR) and the 4-fiber BLSR (4F-BLSR). In 2F-BLSR two fibers in opposite directions are used to interconnect neighboring nodes and half of the capacity on each of these fibers is reserved for failure recovery. In 4F-BLSR two fibers are used to carry the working traffic in the two opposite directions of the ring and the other two fibers are used as backups to be utilized only in the case of a failure. In UPSR, traffic is simultaneously transmitted on the working and protection fibers in the two directions of the ring and the receiver chooses the signal with the best quality.

In mesh network architectures, cycle-based techniques can also be applied, provided that we can find a ring decomposition of the mesh architecture and then use well-established protection switching schemes to restore the traffic whenever a failure occurs. The three most notable ring-based protection techniques for mesh networks are *ring covers*, *cycle double covers*, and *p-cycles* [24].

Given an undirected graph $G(V, E)$ with a set of nodes V and a set of edges E (with loops and multiple edges allowed), a *ring cover* of G is defined as a set of (not necessarily distinct) cycles, $C = (C_1, ..., C_m)$, $m \geq 1$, of G, such that each edge of G belongs to at least one cycle of C. Thus, a ring cover is a set of cycles that cover all the edges of a graph. The discussion in this work is limited to 2-edge connected graphs, since graphs with bridges have no ring cover. For these bridgeless graphs, a ring cover can always be found. The goal of a ring covers approach is to find a set of rings that covers all the network links and use these rings to protect the network against failures, while trying to keep the length of the ring cover as low as possible in order to minimize the redundant capacity required. The *length* of a ring is defined as the number of edges it contains. Thus, given a graph G with ring cover C, the *ring cover length* ($\mathcal{L}(C)$) is the sum of the lengths of all cycles in C. This

approach, however, achieves more than 100% redundancy, since every network edge is used in at least one ring. Fig. 4.3 shows a ring cover decomposition of a network graph where it is illustrated that some links are included in more than one protection cycles resulting in more than 100% redundancy. To solve the problems encountered by the ring cover approach, the cycle double cover technique was introduced, aiming at finding a set of rings that covers all the network links in such a way that the ring cover requires exactly 100% redundancy. According to this scheme protection fibers are decomposed into a family of directed cycles, in such a way that all protection fibers are used exactly once, and in any directed cycle a pair of protection fibers are not used in both directions. In the example illustrated in Fig. 4.4 it is clearly shown that protection cycles do not overlap in any arc in the network, resulting in 100% redundancy when the inner cycles of the plane embedding of the graph are all traced in one direction (counterclockwise) and the outer face is traced in the opposite direction (clockwise). Work in [58] showed that cycle double covers can also be found in networks with planar, as well as non-planar topologies, thus guaranteeing that cycle-based protection in mesh optical networks is always possible with exactly 100% redundant capacity.

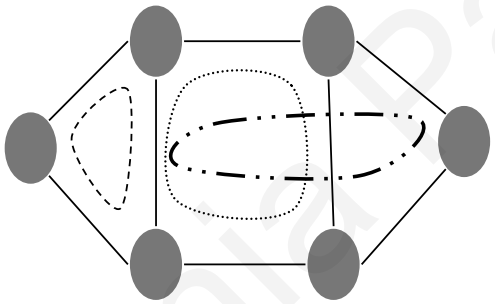

Figure 4.3: Example of a ring cover.

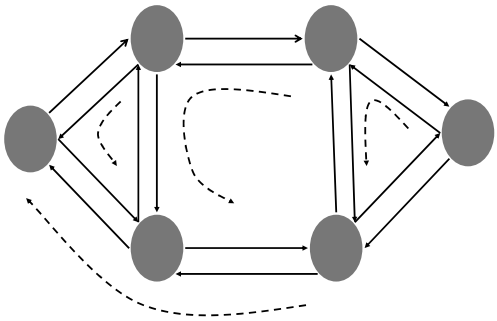

Figure 4.4: Example of a cycle double cover.

Preconfigured cycles (p-cycles) are another set of cycles that can be utilized for protection that try to minimize the redundancy of the network (< 100%) while keeping recovery speeds on the order of a few milliseconds. Compared to conventional ring covers, a p-cycle provides two protection paths for each link that straddles the cycle along with the protection of on-cycle links. Therefore, straddling links can have working capacity but no spare capacity, which is a very unique characteristic of p-cycle-based protection.

Fig. 4.5 shows a p-cycle passing through the nodes (a, d, c, b, f, e), capable of protecting all the links that form the cycle as well as links (e, d) and (a, b) that are the chords of this cycle (the straddling links are just the chords of the cycle, while the on-cycle links are the

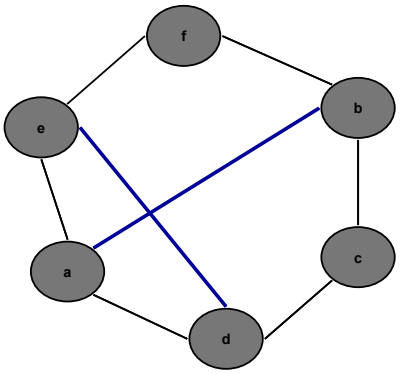

Figure 4.5: Example of a p-cycle with two remaining chords.

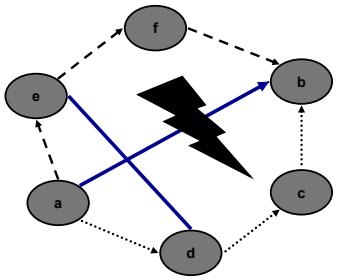

Figure 4.6: Protection paths in case a link failure occurs on a straddling link.

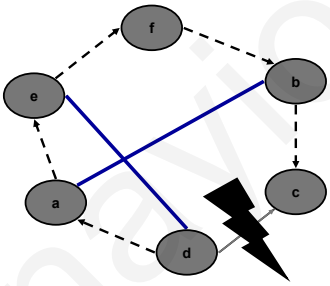

Figure 4.7: Protection path in case a link failure occurs on an on-cycle link.

links that form the protection cycle). As shown in Fig. 4.6, if a failure occurs on a straddling link of the p-cycle, then two protection paths exist for protecting the link. Specifically, if a link failure occurs on link (a, b) then either path (a, d, c, b) or path (a, e, f, b) can be used for protecting the link (a, b). However, if a link failure occurs on an on-cycle link, e.g., link (d, e), then one protection path exists for protecting the arc, as illustrated in Fig. 4.7.

One of the main advantages of the p-cycles approach is that less than 100% redundancy is required to protect the mesh network against any link failure as the straddling links have working capacity but require zero units of protection capacity. p-Cycle-based protection has emerged as a topic of great interest over the past few years due to its capabilities of achieving ring-like high-speed failure recovery with mesh-like high efficiency in the use of spare capacity [66]. A survey on the p-cycle protection method for unicast as well as for multicast traffic can be found in [90].

The idea of optimal spare capacity design for p-cycle-based restorable networks was first formulated in [109, 179] using integer linear programming (ILP). p-Cycle schemes for protecting multicast traffic have only recently started receiving some attention [62,218,219,227].

When designing survivable multicast networks with p-cycles, every link of all multicast trees should be protected by p-cycles, while minimizing the spare capacity required. This would result in ILP formulations which have by far a much higher number of variables and constraints compared to the case where only unicast demands are considered. ILP-based methods to cope with this problem are introduced in [219, 227]. Specifically, in [227] a set of ILP methods was proposed for decomposing the network into the appropriate set of cycles and then a heuristic approach was presented, namely the efficiency ratio heuristic (ERH), for provisioning static multicast sessions. During ERH all multicast sessions were routed, and then the most efficient p-cycles for protecting all the established multicast trees were identified aiming at reducing the total spare capacity reserved for p-cycles. Similarly, in [219] an ILP method was proposed for the configuration of the p-cycles aiming at maximizing the amount of protected working capacity for a given network topology. For the provisioning of dynamic multicast sessions three strategies were proposed; in strategy 1, all the existing p-cycles are released and then reconfigured upon the arrival of a new multicast request, strategy 2 attempts to maximize the number of working units that can be protected by existing p-cycles and reconfigure new ones if the new multicast tree cannot be protected by the existing ones, and in strategy 3 if the routing of a new light-tree fails, it follows strategy 1, otherwise it follows strategy 2. As expected, strategy 1 achieves a better blocking performance while strategy 2 requires much less computational time. Strategy 3 achieves the best blocking performance and its computational time is close to that of strategy 2. In [62] another method called intelligent p-cycle (IpC) is introduced for protecting dynamic multicast sessions and there it is shown that it outperforms the method presented in [219]. During IpC, a multicast tree is computed upon the arrival of a multicast call and then a set of p-cycles is computed to protect the multicast tree. Specifically, the IpC heuristic finds dynamically a set of p-cycles to protect each multicast tree arriving into the network. For every link in the multicast tree there are two options to protect it; either by finding a new p-cycle for it, or extending an existing p-cycle. Among the candidate p-cycles, the p-cycle with the maximum efficiency ratio (ER) is chosen. In that work, the ER of a p-cycle is defined as the ratio of the number of links in the multicast tree that are protected by the p-cycle to the number of links used by the p-cycle.

4.2 Segment-Based Protection Algorithms for Multicast Sessions

As pointed out, protecting a multicast session against a link failure requires finding alternate paths for all failure scenarios prior to the occurrence of the fault. Various approaches exist for multicast protection including path-, segment-, and cycle-based techniques. This thesis investigates the problem of segment-based protection of multicast connections in mesh optical networks, as segment-based protection schemes are reported to have a combined better performance than other known schemes in terms of resource efficiency, recovery speed, and blocking probability. The assumption in this work is that a multicast call is accepted into the network only if a working tree and backup paths can be found that can provide 100% protection of the traffic in the case of a single link failure. The proposed (Level Protection (LP)) heuristic algorithm is compared to different protection algorithms described in the literature [107, 198] and is shown to improve performance especially for the case where the PLIs are taken into account when calculating the primary (working) tree and the backup (protection) paths. The novelty of the work stems from the fact that in the literature most of the work that includes PLIs deals only with unicast connections, whereas in this work a multicast protection algorithm is presented that performs well even when the PLIs are considered.

The proposed LP (level protection) heuristic algorithm differs from the commonly used SP (segment protection) and SSNF (segment-based protection with sister node first) heuristic algorithms discussed in detail below (these are some of the most commonly used segment-based protection approaches that provide very good performance results compared to other protection techniques) in how the segments are identified, as there are many possible ways of dividing the working tree. The objective of this work is to minimize the overall network blocking probability for the multicast connections in the network.

The SP and SSNF protection schemes are described below, slightly modified now to account for directed trees and backup paths.

4.2.1 Modified Segment Protection Heuristic Algorithm

The modified conventional segment protection algorithm (MCSP) described here is a variation of the segment protection with load balancing (SPLB) technique described in [198]. Directed trees and backup protection paths are considered in this approach, instead of considering bidirectional paths. Due to these considerations, the primary segments are now

slightly changed as they are found not only according to the branch nodes [198] but also according to the destination nodes. The steps of the modified SP algorithm are described after some definitions that follow next.

- **Branch point of a tree:** Given a tree T, a vertex $v \in T$ is said to be a branch point if its nodal degree (including the in-degree and the out-degree) is not less than 3 or it is the root.
- **Segment point of a tree:** Given a tree T, a vertex $v \in T$ is said to be a segment point if it is a branch point or is a destination node of the tree.
- **Segments of a tree:** Given a tree T, a path between two segment points is said to be a segment of the tree if the path does not pass through any other segment points of the tree except the two end-nodes of the path.

The main steps of the MCSP heuristic algorithm are given in Algorithm 11 below.

4.2.2 Modified Segment-Based Protection with Sister Node First Heuristic Algorithm

The authors in [107] proposed a segment-based protection scheme called segment-based protection with sister node first (SSNF). Its basic idea is to protect a primary tree using a set of backup segments, with higher priority to protect the segments from a splitting node to its children. In this scheme, sister nodes are defined as the nodes with a common parent and segment points are defined as the splitting nodes or the nodes with degree 1. Initially, a primary tree T is found and an auxiliary graph AG is created consisting of the segment points and the segments of the primary tree T. Once the sister nodes are identified on the AG, the SSNF algorithm tries to protect the segments from branch point to its children using a least-cost tree. If such a least-cost tree does not exist, then the segment protection approach is used to protect each segment individually. The SSNF algorithm includes both inter- and intra-tree sharing because the cost of all links, which are either part of the primary tree or are already used as backup links, is set to 0.

The following definitions are used for the SSNF heuristic:

- **Branch point of a tree:** Given a tree T, a vertex $v \in T$ is said to be a branch point if its nodal degree (including the in-degree and the out-degree) is not less than 3 or it is the root.

Algorithm 11 MCSP

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [v_1, v_2, ...v_n] \subseteq V$, and distance costs $c(e)$ assigned to each arc $e \in E$.
Output: A working tree T spanning the set $s \cup D$ and a set of backup arcs L for the working tree.

1: In G find a minimum cost tree T spanning the source node s and the destination node set D using the Steiner Tree (ST) heuristic.
2: Identify the segment points in tree T. Identify the segments in tree T. Let k be the total number of primary segments.
3: $L \leftarrow$ Empty set.
4: $i \leftarrow 1$
5: **while** $i \leq k$ **do**
6: Generate an auxiliary graph G' as follows:

1. Add all the nodes in the initial graph G into G'.

2. For each arc e in G, if the arc is neither in the primary light-tree T nor in set L, add the arc into G' and set for it an arc cost $c(e)$. If the arc is in primary tree T but not in primary segment i, add the arc into G' and set a cost 0 for the arc. Similarly, if the arc is in set L but not in primary segment i, add the arc into G' and set an arc cost 0 for the link.

7: Compute in G' the shortest path p between the two segment points of segment i using Dijkstra's algorithms and considering as source node the segment point closer to the source node $s \in V$.
8: $L \leftarrow p$
9: $i \leftarrow i + 1$
10: **end while**
11: **return** T and L

- **Segment point of a tree:** Given a tree T, a vertex $v \in T$ is said to be a segment point if it is a branch point or it is a destination node.

- **Segments of a tree:** Given a tree T, a path between two segment points is said to be a segment of the tree if the path does not pass through any other segment points of the tree except the two end-nodes of the path.

- **Sister nodes in an auxiliary graph with respect to tree T:** Given an auxiliary graph AG with respect to tree T, two nodes in the auxiliary graph AG are said to be sister nodes if they have a common parent.

The SSNF algorithm is described in detail in [107]. Here, only the main steps of the algorithm are shown in Algorithm 12, with some modifications so as the algorithm can be adjusted when directed primary trees are considered and also directed backup paths for protecting the primary segments of the tree are derived (modified segment-based protection with

sister node first (MSSNF) heuristic algorithm). In [107] the primary tree is undirected and so both sides of a link are assumed to construct both primary and backup paths. Note that $C(v)$ is the set of nodes, each of which is a child of the segment point v.

Algorithm 12 MSSNF

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [v_1, v_2, ...v_n] \subseteq V$, and distance costs $c(e)$ assigned to each arc $e \in E$.
Output: A working tree T spanning the set $s \cup D$ and a set of backup arcs L for the working tree.

In G find a minimum cost tree T spanning the source s and the destination set D using the Steiner Tree (ST) heuristic.
Identify the segment points in tree T. Identify the segments in tree T.
Generate an auxiliary graph AG with respect to the primarily tree T according to its segment points and segments.
Identify sets $C(v)$ in AG. Let k be the total number of sets $C(v)$ in AG.
$L \leftarrow$ Empty set.
$i \leftarrow 1$
while $i \leq k$ **do**
    Generate an auxiliary graph G' as follows:

    1. Add all the nodes in the initial graph G into G'.

    2. For each arc e in G, if the arc is neither in the primary tree T nor in set L, add the arc into G' and set for it an arc cost $c(e)$. If the arc is in primary tree T but not in any segments from node v to its children, add the arc into G' and set a cost 0 for the arc. Similarly, if the arc is in set L but not in any segments from node v to its children, add the arc into G' and set a cost 0 for the arc.

    if the number of nodes in $C(v) > 1$ **then**
        Derive in G' a least-cost tree p that spans all the nodes in set $C(v)$.
        if a least-cost tree p exists **then**
            $L \leftarrow p$ *(Note that although the tree is directed, both sides of the least-cost tree that spans the nodes in set* $C(v)$ *have to exist for protection purposes, and so both sides of the links are added into set* L*.)*
            $i \leftarrow i + 1$
        end if
    end if
    if (The number of nodes in $C(v) = 1$) OR (The number of nodes in $C(v) > 1$ but a least-cost tree does not exist) **then**
        For each node in $C(v)$ use segment protection to protect each segment separately.
        Add in L the calculated shortest paths for each node in $C(v)$. *(Note that only the arcs in the direction of the calculated path have to be added into set* L*.)*
    end if
end while
return T and L

The section that follows describes a novel segment-based multicast protection heuristic algorithm called the level protection (LP) heuristic algorithm. It differs from the other

segment-based protection approaches in the way the segments are identified. This technique is shown to have improved performance when the physical impairments are also taken into consideration, which was precisely the goal when designing this protection technique.

4.2.3 Level Protection Heuristic Algorithm

The key idea behind the Level Protection (LP) heuristic algorithm is initially described using Figs. 4.8- 4.11. In Fig. 4.8 a general tree is shown in which each node is a destination node, apart from the root node which is the source. The levels of every group of nodes is also illustrated in Fig. 4.8. Fig. 4.9 shows that if a directed protection path exists that starts from a level 0 node and spans every node in level 1, then upon any link failure between levels 0 and 1, information from level 0 node can still reach every destination node in level 1. Similarly, Fig. 4.10 shows that since information from the source can definitely reach nodes in levels 0 and 1, either via the primary paths or via the protection paths, then if a directed path can be found originating at any node in levels 0 and 1 and spanning nodes in level 2, then the information from the source can still reach every node in level 2 in case of a link failure. By the same way, Fig. 4.11 shows the protection paths between levels 2 and 3 which ensure that information from the source can still reach every node in level 3 in the event of a link failure. Since information from the source can reach every node in levels 0, 1, and 2, then a directed path starting from any node in higher levels and spanning every node in level 3 can protect every link between levels 2 and 3. Note that the level protection scheme can support more than one link failure with the constraint that all the link failures must happen between the same levels of the tree.

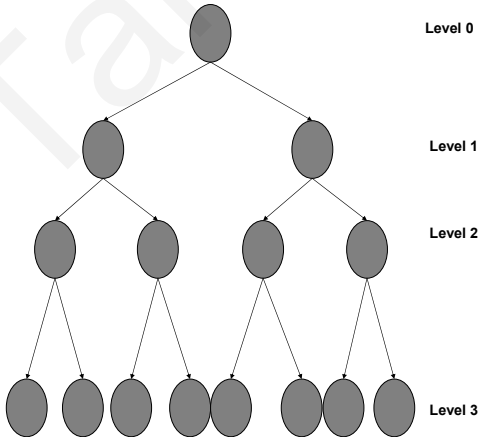

Figure 4.8: General tree with the levels for each node shown.

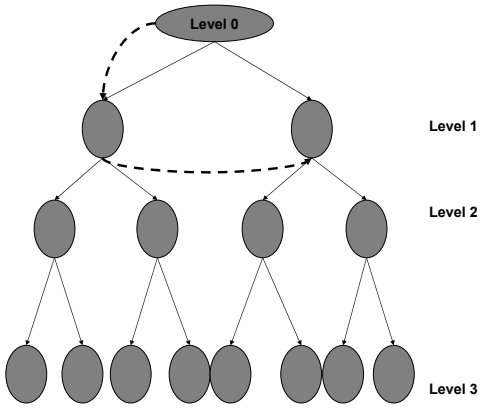

Figure 4.9: General tree with protection paths between levels 0 and 1.

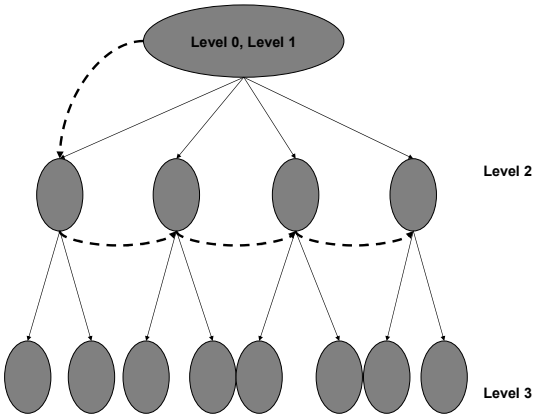

Figure 4.10: General tree with protection paths between levels $0/1$ and 2.

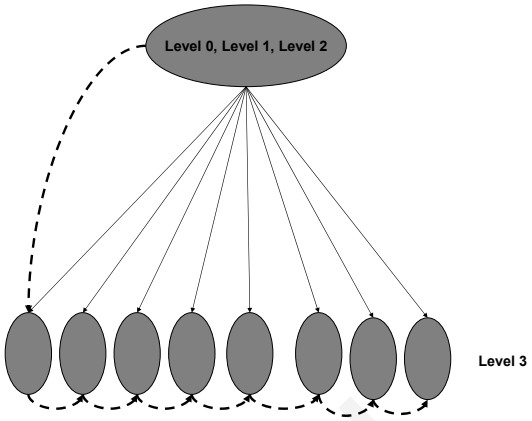

Figure 4.11: General tree with protection paths between levels $0/1/2$ and 3.

In order to explain the LP heuristic algorithm some definitions are first needed:

- **Segmentation nodes of a tree:** Given a tree T, a vertex $v \in T$ is said to be a segmentation node if it is the source s or it is a destination node of the multicast request. Consider the tree T shown in Fig. 4.12 as an example, where node s is the source node and also a segmentation node. Since nodes d_1, d_2, d_3, d_4, and d_5 are the destination nodes of the tree, they are also segmentation nodes of the tree.

- **Segments of a tree:** Given a tree T, a path between two segmentation nodes is said to be a segment of the tree if the path does not pass through any other segmentation node of the tree, except the two end-nodes of the path. For example, the path $s - u - d_1$ in Fig. 4.12 is a segment. Also, paths $d_1 - d_2$, $d_2 - d_5$, $d_2 - d_4$, and $d_4 - d_3$ are segments of tree T.

- **Level$_i$ of a segmentation node:** Given a tree T, the level value for a segmentation node is the number of the segments between that segmentation node and the source s. Therefore, segmentation node s in Fig. 4.12 is at level zero (denoted by Level$_0$). Fig. 4.13 shows an auxiliary graph AG, created based on the segmentation nodes of the primary tree, where level values for the segmentation nodes are also demonstrated. For example, segmentation node d_1, has level value 1. Similarly, segmentation node d_2 belongs to Level$_2$, segmentation nodes d_4 and d_5 to Level$_3$ and segmentation node d_3 to Level$_4$. Auxiliary graph AG of Fig. 4.13 has a maximum depth of four levels.

- **Level$_{(i,i+1)}$ segment group:** Given a tree T, level$_{(i,i+1)}$ segment group is the set of all the segments of the tree that lie between the segmentation nodes of Level$_i$ and the

segmentation nodes of Level$_{i+1}$. For example, the set of segments $d_2 - d_4$ and $d_2 - d_5$ of Fig. 4.13 is a Level$_{(2,3)}$ segment group. Similarly, segment $s - d_1$ is a Level$_{(0,1)}$ segment group, segment $d_1 - d_2$ is a Level$_{(1,2)}$ segment group and segment $d_4 - d_3$ is a Level$_{(3,4)}$ segment group.

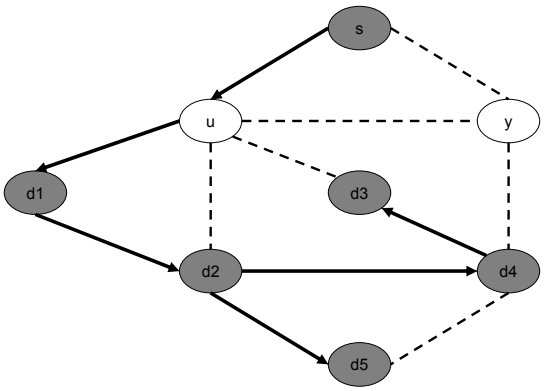

Figure 4.12: Primary tree T for a multicast request with source node s and destination set $[d_1, d_2, d_3, d_4, d_5]$.

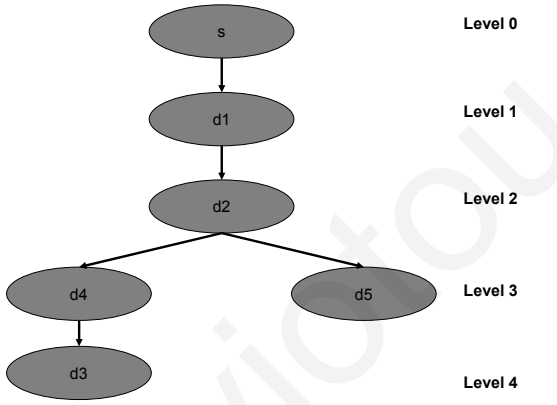

Figure 4.13: Auxiliary graph AG based on the segmentation nodes of the primary tree T with the level values for each segmentation node shown.

Segments in the same group are protected by computing a single arc-disjoint backup path that starts from any lower-level segmentation node and spans every segmentation node in the same group. The protection process of the LP heuristic starts from the lower levels and continues to the higher levels in a hierarchical manner. Therefore, backup paths have to be arc-disjoint from their segment group and arc-disjoint from the segment groups that lie between higher levels. Self- and cross-sharing are also considered with this approach in order to improve the resource utilization ratio.

Specifically, the proposed LP heuristic algorithm works as follows: Once a multicast request arrives into the system, the algorithm first calculates the primary tree T on graph G using the Steiner tree heuristic and then for all the arcs of graph G that belong to tree T it updates their weights to a zero value. Next, the algorithm identifies the segmentation nodes, the segments, the level i values on the auxiliary graph, and the $L(i, i+1)$ segment groups as defined above. Once the $L(i, i+1)$ segment groups are identified, i is initialized to zero value and k is assigned the level value of the leaf nodes on the auxiliary graph. The algorithm starts the protection procedure for the $L(i, i+1)$ segment group by calculating all backup paths that start from every segmentation node that belongs to Level j, where $0 \leq j \leq i$, and span the segmentation nodes that belong to Level $(i+1)$. The minimum cost backup path amongst

them is then chosen. The algorithm keeps incrementing i by 1 and the protection procedure terminates when $i = k - 1$. Note that the backup paths of the $L(i, i+1)$ segment group are calculated using Dijkstra's shortest path algorithm if the number of segmentation nodes in Level $(i+1)$ is less than two; otherwise it is calculated using the Steiner tree heuristic. To improve the resource utilization in the network, each time a backup path is chosen, the arc weights on graph G are updated to zero for all the arcs in G that are used for the construction of the backup path.

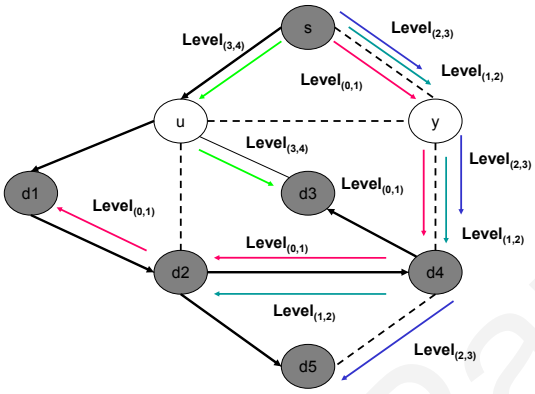

Figure 4.14: Protection paths for each level segment group.

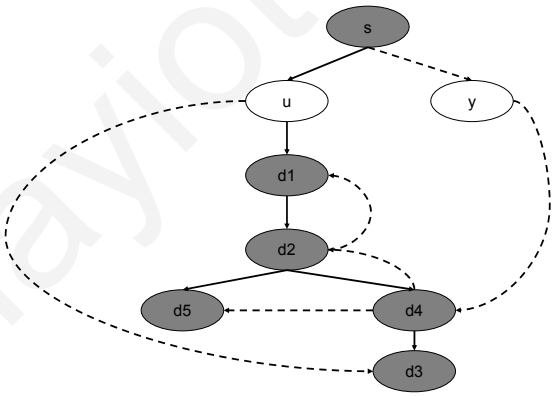

Figure 4.15: Combined primary tree and backup paths.

Figs 4.14 and 4.15 are used to illustrate the operation of the LP heuristic algorithm. Once the primary tree, segmentation nodes, segments, levels on the auxiliary graph, and $L(i, i+1)$ segment groups are identified, backup paths are determined starting from the lower levels. For example, in Fig. 4.14, for the calculation of the $L(0, 1)$ backup path, the algorithm uses the shortest path heuristic starting from node s at Level 0 and moving to the d_1 node of Level 1. Fig. 4.14 shows the backup paths found for each $L(i, i+1)$ segment group (sharing is clearly illustrated). In Fig. 4.15, primary tree T and backup paths are combined together and it is shown that in case of any single link failure, data from the source can still reach every destination node on the multicast tree.

The main steps of the LP heuristic are shown in Algorithm 13 below.

Algorithm 13 LP

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [v_1, v_2, ...v_n] \subseteq V$, and distance costs $c(e)$ assigned to each arc $e \in E$.
Output: A working tree T spanning the set $s \cup D$ and a set of backup arcs L for the working tree.

1: In G calculate a minimum cost tree T spanning the source s and the destinations set D using the Steiner Tree heuristic.
2: Identify the segmentation nodes in tree T. Identify the segments in tree T.
3: Generate an auxiliary graph AG with respect to the primary tree T according to the segmentation nodes and the segments of tree T.
4: For each segmentation node in AG identify its level value. Let $l(i)$ be the set of segmentation nodes that lie in the same level.
5: $L \leftarrow$ Empty set.
6: Identify the maximum level k of tree T.
7: $i \leftarrow 0$.
8: **while** $i < k$ **do**
9: Generate an auxiliary graph G' as follows:

1. Add all the nodes of the initial graph G into G'.

2. For each arc e in G, if the arc is neither in the primary tree T nor in set L, add the arc into G' and set for it a cost $c(e)$. If the arc is in the primary tree T but not in the $L(i-1, i)$ segment and in any level segment that is below $L(i-1, i)$ segment, add the arc into G' and set a cost 0 for the arc. Similarly, if the arc is in set L but not in the $L(i-1, i)$ segment and in any level segment that is below $L(i-1, i)$ segment, add the arc into G' and set a cost 0 for the arc. Note that by saying below $L(i-1, i)$ segment, it is meant that $L(i, i+1)$ to $L(k-1, k)$ segment arcs are not added into G' *(Note that k is the depth of the tree)*.

10: For each segmentation node in $l(i-1)$, derive in G' a minimum cost tree starting from that segmentation node and spanning all the nodes in set $l(i)$.
11: Choose the minimum cost tree p amongst them.
12: $L \leftarrow p$.
13: Add in set $l(i)$ all the segmentation nodes of set $l(i-1)$.
14: $i \leftarrow i + 1$
15: **end while**
16: **return** T and L

4.3 Cycle-Based Protection Algorithms for Multicast Sessions

This work investigates the problem of protection for multicast connections in transparent optical networks utilizing a p-cycles approach, since as previously mentioned the p-cycle-based protection techniques have emerged as suitable candidates for protection against link failures in current network deployments due to their capabilities of achieving ring-like recovery speed, with mesh-like efficiency in the use of spare (redundant) capacity [90]. Although

many p-cycle-based schemes have been proposed for unicast protection, applying p-cycles for multicast protection under PLIs has barely been studied.

Thus, in this section a Q-based p-cycles heuristic algorithm is developed, namely the QBPCH algorithm, which decomposes the network into a set of p-cycles by setting a constraint on the maximum length of the p-cycles, utilizing the Q-budgeting approach described in Chapter 2 to account for the physical layer impairments. By doing this, the QBPCH algorithm aims at minimizing the impact of the physical layer impairments upon the creation of the p-cycles, and hence the blocking probability for the multicast sessions. A second p-cycles heuristic algorithm is also proposed, namely the PCH algorithm, which aims at maximizing the length of the p-cycles and thus the number of straddling links covered by the p-cycles, thus minimizing the overall redundant capacity required in the network.

The QBPCH and PCH heuristics are developed and compared with a conventional ring-cover approach denoted as the RC heuristic and with a Hamiltonian Cycle (HC) approach. The RC heuristic decomposes the network into a set of fundamental cycles in such a way that every link in the network belongs to at least one cycle. For graph G, a ring cover can be obtained by finding the set of *fundamental cycles* for the graph. The set of fundamental cycles forms a basis for the cycle space, and any arbitrary cycle of the graph can be expressed as a linear combination of the fundamental cycles using the ring-sum (exclusive or) operation. A *Hamiltonian cycle* of a graph G is a cycle which contains all the vertices of G and passes through every node in the network exactly once. For both the RC and HC schemes, several algorithms have been presented in the literature. Specifically for the RC approach different techniques can be found in [24] and references therein. The basic steps of the RC heuristic used in this work for finding a set of ring covers for a graph G are given in Algorithm 14. Algorithms for finding Hamiltonian cycles can be found in [193]. Note that for the performance evaluation part network graphs were used where it was always possible to find Hamiltonian cycles. By comparing the proposed p-cycle heuristics to the above approaches the PCH/QBPCH heuristics are compared to the two possible extreme cases; those of the minimum and maximum (in length) protection cycles in the network.

Unlike other protection approaches like tree- and segment-based in which the backup paths for a multicast call are dynamically computed upon the arrival of the multicast request, when cycle-based schemes are assumed only the primary light-tree has to be dynamically computed upon the arrival of the call. The creation of backup paths corresponds to just merging together all the backup paths that were pre-assigned to each arc involved in the creation of the primary tree. This implies that after the decomposition of the network into a

Algorithm 14 RC Heuristic

Input: A graph $G = (V, E)$ representing the network with equal weights $w_i = 1$ for each link $e_i \in E$, where $i = 1, 2, ..., n$, and a value $h >> 1$.
Output: A set of cycles P.

```
1: i ← 1
2: while i ≤ n do
3:    if w_i = 1 then
4:       In G, find shortest path p_i connecting the end points of link e_i using Dijkstra's
         algorithm.
5:       In G, set w_j = h for each link e_j ∈ p.
6:       In G, set w_i = h.
7:       Merge link e_i and p_i into cycle c_i
8:       Add c_i into set of cycles P.
9:    end if
10:   i ← i + 1
11: end while
12: return P
```

set of cycles, each arc is assigned a backup path based on the precomputed set of cycles. For the backup path assignment procedure, a uniform methodology is followed for every cycle-based scheme developed. The need for developing such a procedure arises due to the fact that an arc may be included in more than one cycles or due to the fact that an arc might be a straddling arc of a cycle. Thus, the minimum cost cycle (MCC) scheme is developed that is used for the assignment of one backup path for each arc in the network. Specifically, MCC chooses for each arc on the network the minimum cost cycle amongst the possible cycles that can support the arc. Note that here the cost on each arc represents its physical distance, which subsequently results in an improvement of the Q-factor. The basic steps of the MCC scheme are given in Table 4.8.

Based on the discussion above, once a multicast call arrives into the network, the primary light-tree T_w is computed according to the Steiner tree (ST) heuristic. Backup paths T_p are then all the paths already assigned to each arc in tree T_w. Note that self-sharing of the resources is possible since the arcs used for the creation of the primary tree can be reused as backup paths.

Table 4.8: Minimum Cost Cycle (MCC) Assignment

Step 1	For each arc (i, j) in network G find the set of cycles C in which the arc is a part of.
Step 2	Extract from set C all backup paths of arc (i, j).
Step 3	Choose the minimum cost path amongst them.

4.3.1 p-Cycle Heuristic (PCH) Algorithm

The PCH heuristic algorithm aims at breaking an arbitrary network G into a set of cycles in such a way that the number of the straddling links is maximized. The key idea is to create these cycles starting from a random p-cycle in the network and then extending this p-cycle by sequentially replacing those links in the cycle for which alternate paths exist. The constraint is that the new paths do not pass though the nodes that are already added to the cycle. For example, if such a path exists for a certain link, that link is removed from the cycle and the new path is added. The link removed is now declared to be a straddling link of the cycle. The heuristic is repeated until every link in the p-cycle is checked and cannot be replaced by an alternate path. Upon the termination of this procedure, the heuristic seeks to find if there are any links left in the network, that are neither on-cycle nor straddling links of the p-cycle. If such links exist, a new graph G' is created that consists of all the nodes and links of the initial graph G apart from the straddling links of the p-cycle. Weights on graph G' are set in such a way that the weights of the on-cycle links of the p-cycle are assigned a value that is several orders of magnitude greater than the weights of the rest of the links in G'. By doing so the reutilization of p-cycle links is minimized. After the creation of graph G', the RC heuristic described in Algorithm 14 is used for the creation of the rest of the cycles.

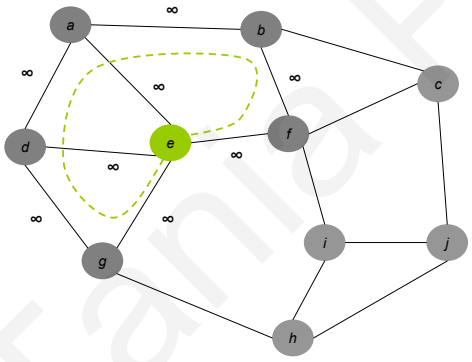

Figure 4.16: Creation of the first p-cycle.

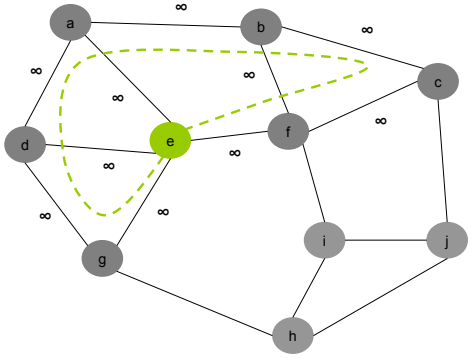

Figure 4.17: Extension of the first p-cycle.

Specifically, the PCH algorithm creates a main cycle for which the number of straddling links is sequentially increased. For the links not included in the main p-cycle, minimum cost cycles are created for their protection. The first p-cycle is created starting initially from the maximum degree node in the network. Thus, the PCH heuristic first calculates set D consisting of the degrees for each node in graph G and then identifies the maximum degree node m as well as the $m(v)$ set consisting of the nodes that are adjacent to m. It subsequently builds the first p-cycle starting from node m and passing through every node in set $m(v)$

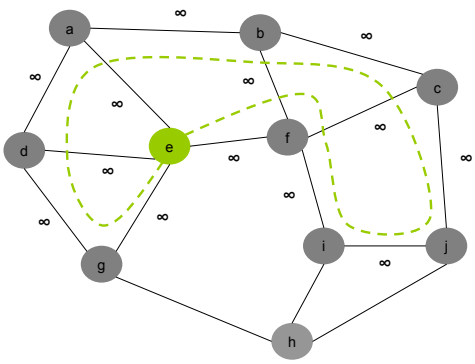

Figure 4.18: Extension of the p-cycle of Fig. 4.17.

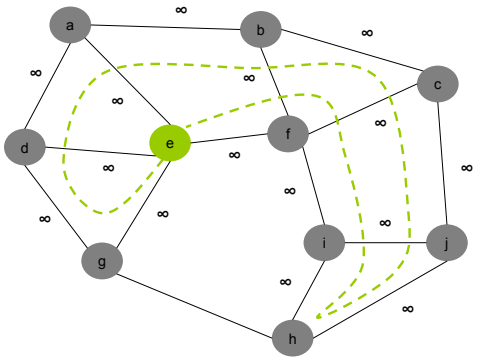

Figure 4.19: Extension of the p-cycle of Fig. 4.18.

only once. Fig. 4.16 shows the first p-cycle created on a network topology, starting from the maximum degree node e and passing through its adjacent nodes $m(v) = [g, d, a, f]$ only once. Note that links directly connected to node m but not included in the construction of the p-cycle are the straddling links of that p-cycle. For the construction of the first p-cycle, the Optimized Collapsed Ring (OCR) heuristic proposed in [88] and described in Algorithm 15 is used. To extend the first p-cycle, the weights of the graph are updated in such a way that both on-cycle and straddling links of the already constructed p-cycle are set to an infinite value in order to be excluded from the extension procedure. PCH then seeks to find the first link in the cycle that can be removed and replaced by an alternate path. According to Fig. 4.16 and Dijkstra's algorithm, link (e, f) cannot be replaced by any alternate path but link (f, b) can be replaced by path $[(f, c), (c, b)]$. As shown in Fig. 4.17, path $[(f, c), (c, b)]$ replaces link (f, b) and the weights of the new links added to the p-cycle are set to an infinite value. Following the same procedure, link (f, c) is replaced by path $[(f, i), (i, j), (j, c)]$ as shown in Fig. 4.18 and then link (i, j) is replaced by path $[(i, h), (h, j)]$ as shown in Fig. 4.19.

The extension of the main p-cycle terminates when no other replacement can be made. After the termination of the extension procedure, PCH seeks to find if there are any links left out of the main p-cycle. Links with a weight value less than the infinite value are all the links that were not included in the p-cycle. If such links exist, PCH creates a new graph G' by copying initial graph G to graph G' and removing from G' the straddling links of the main p-cycle. Note that links that are not included in the main p-cycle, but their endpoints lie within the main p-cycle, are also removed from graph G'. The weights of graph G' are then adjusted in such a way that on-cycle links of the main p-cycle are set at a value h that is several orders of magnitude greater than the original link weights. Then the RC heuristic is performed on graph G'.

Algorithm 15 OCR Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $m \in V$, a destination set $m(v) = [d_1, d_2, ...d_n] \subseteq V$, and equal weights assigned to each edge $e \in E$.
Output: A cycle C starting from node $m \in V$ and passing though every node in set $m(v)$ just once.

1: begin
2: $C_1 \leftarrow m$
3: $k \leftarrow 0$
4: $s \leftarrow m$
5: **while** $k \leq n$ **do**
6: Find set P consisting of all shortest paths from node s to every destination node $\in m(v)$.
7: Identify shortest path p amongst the paths $\in P$.
8: Add p to C_1
9: Identify node d_j last added to C_1.
10: Remove destination node d_j from $m(v)$.
11: $s \leftarrow d_j$
12: $k \leftarrow k + 1$
13: **end while**
14: In G, calculate shortest path C_2 from node s to node m, without passing through any other node $\in C_1$ apart from node m.
15: Merge paths C_1 and C_2 into cycle C
16: **return** C

Returning back to the example of Fig. 4.19, the endpoints of the remaining (g, h) link lie within the main p-cycle and thus link (g, h) is also a straddling link of the main p-cycle. Therefore, PCH terminates by creating a single p-cycle with 6 straddling links. In this example, a Hamiltonian cycle is created upon the termination of PCH heuristic. Note, however, that PCH does not always find the Hamiltonian cycle of a random graph even if the Hamiltonian cycle exists. The basic steps of the PCH heuristic for a given network G are shown in Table 4.9.

4.3.2 Q-Based p-Cycle Heuristic (QBPCH) Algorithm

As previously pointed out, the QBPCH technique decomposes an arbitrary mesh network into a set of p-cycles while at the same time it controls the length of the p-cycles according to a predetermined Q-threshold. The length of the cycles is an important parameter since constructing the fewest p-cycles may be more efficient in terms of capacity but the size of a cycle even if is not a single Hamiltonian cycle can be very large [125]. In general, p-cycles that are considerably large in size are especially not desirable because they would lead to extremely long backup paths, which in turn may lead to unacceptable signal quality

Table 4.9: Basic Steps of the PCH Heuristic

Step 1	Copy graph G to G'.
Step 2	In G', identify maximum degree node m and its adjacent nodes $m(v)$.
Step 3	Evaluate p-cycle C_1 according to the OCR heuristic of Algorithm 15.
Step 4	Let link (m, i) be the first link in C_1 and link (j, m) be the last link added in C_1.
Step 5	Update link weights on graph G' by setting an infinite cost to every on-cycle and straddling link included in C_1.
Step 6	Starting from link (m, i), identify the first link (u, v) in C_1 for which an alternate path $p_{u,v}$ exists between its endpoints. Calculate $p_{u,v}$ using Dijkstra's algorithm on graph G' between nodes u and v.
Step 7	In C_1, replace link (u, v) by p_{uv}.
Step 8	Repeat Steps 5 to 6 for up to link (j, m).
Step 9	Copy graph G to G'' and remove from G'' all straddling links of C_1.
Step 10	Remove from G'' the links that their endpoints are attached to C_1.
Step 11	Update link weights on graph G'' as follows: • For every on-cycle link of C_1 set a link weight equal to h. • For every other link on graph G'' set link weights equal to f, such that $h >> f$.
Step 12	In G'', evaluate set of cycles C_2 according to the RC heuristic of Algorithm 14.
Step 13	Return set of cycles C_1 and C_2.

at the receiver [90]. However, the decomposition of the network into the smallest (in size) possible p-cycles may not be capacity efficient. Thus, a threshold based on appropriate PLIs is considered in the QBPCH algorithm during the construction phase of the p-cycles.

Specifically, the QBPCH heuristic aims at breaking an arbitrary network G into a set of cycles in such a way that the length of the cycles meet the PLI constraints. The key idea is to create these cycles starting from the maximum degree node in the network. Thus, the QBPCH heuristic first calculates set D consisting of the degrees for each node in graph G and then identifies the maximum degree node m as well as the $m(v)$ set consisting of the nodes that are adjacent to m. It then builds the first p-cycle starting from node m and passing through every node in set $m(v)$ only once. Note that links directly connected to m but not included in the construction of the cycle are the straddling links of that cycle. For the construction of the p-cycles, a modified OCR (MOCR) heuristic which is an extension of the OCR heuristic proposed in [88] and described in Algorithm 15 is used. The difference between the MOCR and OCR heuristics is that MOCR considers also the PLIs during the creation of each p-cycle. After every node in set $m(v)$ is connected to node m via a linear path C_1, the Q-factor of each node in set $m(v)$ is evaluated on C_1. Considering a Q-threshold

q, MOCR identifies the last node $d_i \in m(v)$ added to path C_1, with a Q-value that is above q. Subsequently, if $d_j \in m(v)$ is the last node added to path C_1, the algorithm removes from C_1 the path that lies between nodes $d_i - d_j$. Thus, a p-cycle is created between nodes m and d_i. For example, in Fig. 4.20 a linear path C_1 is created between node m and nodes in $m(v)$, where $m = E$ and $m(v) = [C, B, D, G]$. According to the MOCR heuristic, the Q-factor is evaluated for nodes C, B, D, and G starting from node E. Then, assuming that the algorithm identifies that node D is the last node added to C_1 with a Q-value above the Q-threshold q, path (D, F, G) is removed from C_1, and a cycle closes between nodes D and E as shown in Fig. 4.21.

The MOCR heuristic for a node m, a set $m(v)$ consisting of k nodes, and a Q-threshold q is described in Algorithm 16.

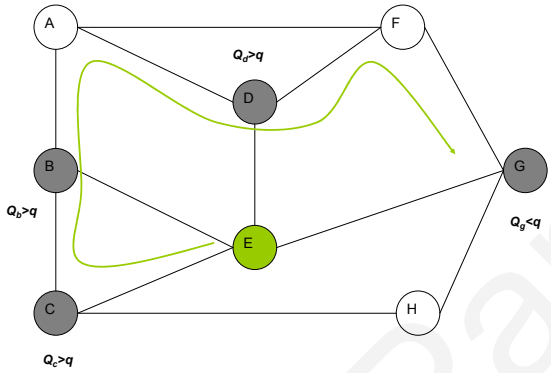

Figure 4.20: MOCR example: Once the linear path is constructed between the maximum degree node m and its adjacent nodes $m(v)$, the Q-factor for each node in set $m(v)$ is evaluated.

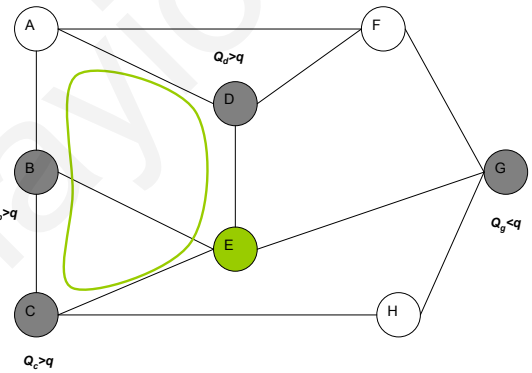

Figure 4.21: MOCR example: The last node added in the linear path with a Q-factor that is above q, is identified and a cycle is created between that node and the maximum degree node m.

After the creation of the first cycle, link weights in graph G are updated in such a way that the weights of the straddling links on the new cycle are set to an infinite value, in order to be excluded from the computation procedure of the p-cycles that will be calculated next, while the weights of on-cycle links are set to a value h which is several orders of magnitude greater than the original link weights. By doing so, the probability of reusing on-cycle links is decreased. To identify the next maximum degree node, the set of node degrees D is calculated based on degree graph G_d that is created according to Table 4.10. The heuristic is repeated until all node degrees in G_d become zero. In case the maximum node degree in G_d is one, the algorithm first examines whether the remaining link is a straddling link of an already computed p-cycle. If it is, then the heuristic terminates. Otherwise a new cycle between the endpoints of the link is created.

Algorithm 16 MOCR Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $m \in V$, a destination set $m(v) = [d_1, d_2, ...d_n] \subseteq V$, equal weights assigned to each edge $e \in E$, and a Q-threshold q. Physical distances between the links are also available for the calculation of the Q-factor.
Output: A cycle C starting from node $m \in V$ and passing though every node in $m(v)$ only once, such that their Q-factor is greater than or equal to q.

1: begin
2: $C_1 \leftarrow m$
3: $k \leftarrow 0$
4: $s \leftarrow m$
5: **while** $k \leq n$ **do**
6: Find set P consisting of all shortest paths from node s to every destination node $\in m(v)$.
7: Identify shortest path p amongst the paths $\in P$.
8: Add p to C_1
9: Identify node d_j last added to C_1.
10: Remove destination node d_j from $m(v)$.
11: $s \leftarrow d_j$
12: $k \leftarrow k + 1$
13: **end while**
14: Calculate the Q-factor of every node $\in D$ starting from node m.
15: **if** the Q-factor of at least one node in D is below q, **then**
16: Identify the last node $s' \in C_1$ with a Q-factor above q such that $s' \in D$.
17: Remove from path C_1 every link and node after node s'.
18: $s = s'$.
19: **end if**
20: In G, calculate shortest path C_2 from node s to node m, without passing through any other node $\in C_1$ apart from node m.
21: Merge paths C_1 and C_2 into cycle C.
22: **return** C

Figs. 4.22- 4.24 are used as an illustrative example of the QBPCH heuristic. For this example it is assumed that the Q-factor for each node in every cycle created is above the predetermined Q-threshold q. Fig. 4.22 shows the network topology in which all link weights are assumed to be equal. Also, the set of node degrees D is illustrated. Node $m = e$ is the maximum degree node with $d_e = 4$. Adjacent nodes to node m are given in $m(v) = \{b, d, g, h, f\}$. Fig. 4.23 illustrates the first p-cycle $C = [c, b, a, d, g, h, i, f, e]$ created according to the OCR heuristic with links $[(e, d), (e, h)]$ being straddling links of cycle C. After the creation of the first p-cycle, the weights of the links are updated depending on whether a link is an on-cycle link, in which case its weight goes to h, or it is a straddling link, in which case its weight goes to infinity. Set D is also updated according to Table 4.10. Both updated link weights and set D are illustrated in Fig. 4.23, where it is shown that node $m = c$ is now the maximum degree node with $m(v) = \{b, j, f\}$. Fig. 4.24 shows the next p-cycle, along with

Table 4.10: G_d Graph for the Calculation of Node Degrees D

Step 1	Add in G_d all nodes of graph G.
Step 2	Add in G_d a link between two nodes if the corresponding link in G has a weight value that is less than h.

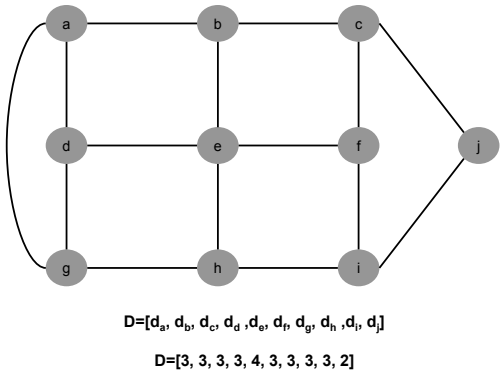

Figure 4.22: Initial set D of node degrees.

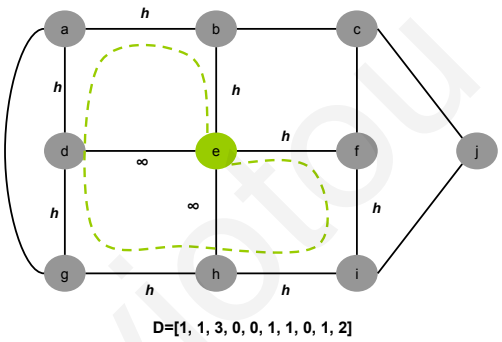

Figure 4.23: Creation of the first p-cycle.

the updated link weights and set D. As shown in Fig. 4.24, now the maximum node degree is one and hence QBPCH seeks to find if the remaining link (link (a, g)) is a straddling link of an already created p-cycle. In this example, the endpoints of the remaining link belong to the same p-cycle, thus link (a, g) is indeed a straddling link of this p-cycle. In this example QBPCH terminates after two iterations, creating 2 p-cycles and having 4 straddling links.

Depending on the Q-threshold that is set in the MOCR heuristic, QBCPH returns a different set of p-cycles. If the Q-threshold is set to a value $q = -\inf$ then QBPCH returns the minimum (in number) set of p-cycles and hence the largest (in length) that QBPCH can create, since the Q-factor of every node in the new cycle will be always greater than q. In the other extreme, if $q = \infty$ then QBPCH returns the minimum (in length) and thus the maximum (in number) cycles. Note that if the Q-threshold is set somewhere between the two extreme values, a different set of cycles is computed and in general the length of the p-cycles is decreased with increasing q. The basic steps of the QBPCH heuristic are given in Table 4.11 for a graph G and a value h that is several orders of magnitude greater than the original link weights.

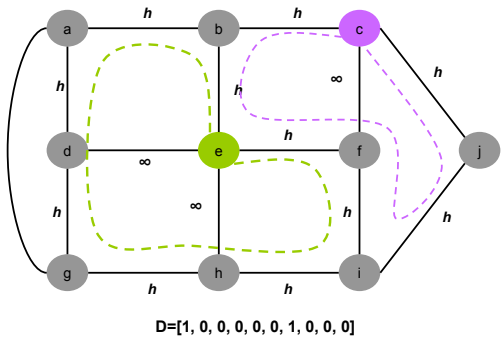

Figure 4.24: Creation of the second p-cycle.

4.4 Provisioning of Protected Multicast Connections

4.4.1 Impairment-Unaware Provisioning

For each multicast request, the multicast routing and wavelength assignment (MC-RWA) algorithm finds a primary light-tree (using the Steiner Tree (ST) heuristic) and its backup paths using any of the aforementioned protection techniques. The primary tree and backup paths must be on the same wavelength and the wavelength assignment is performed using an appropriate wavelength assignment algorithm (in this work the *first-fit* wavelength assignment algorithm is used throughout). It is assumed that optical splitters are present in each node, all connections are directional, there are sixty-four (64) wavelengths per fiber, and there are no wavelength converters in the network. Note that the number of wavelengths is increased in the protection simulations, as more bandwidth is now required for the protected multicast sessions. Since what is of interest in this chapter is the effect of the PLIs on the heuristics and the system performance, the number of available wavelengths was increased so that the blocking probability will not be limited by the number of the wavelengths.

If a primary light-tree T can be provisioned and can also be protected (routing and wavelength assignment for both the primary tree and protection paths is successful), the multicast request is accepted in the network; otherwise it is blocked [129]. Fig. 4.25 shows the flowchart for the MC-RWA algorithm that is used in this work.

4.4.2 Impairment-Aware Provisioning

If PLIs are also taken into account, an impairment-aware protected multicast routing and wavelength assignment (IA-PMC-RWA) algorithm is implemented that ensures that both working and protection paths can deliver acceptable signal quality to the destination nodes.

Table 4.11: Basic Steps of the QBPCH Heuristic

Step 1	Create graph G_d as described in Table 4.10.
Step 2	Create set D representing the degree for each node in G_d.
Step 3	In D, identify node m with maximum degree k.
Step 4	In G, identify set of nodes $m(v)$ directly connected to node m.
Step 5	If: • $k = 0$, then go to Step 9. • $k = 1$, then check if the endpoints of link $(m, m(v))$ belong to a cycle c_j previously constructed. If it does, add link $(m, m(v))$ to the set of straddling links s_j, otherwise create a new cycle according to the OCR heuristic of Algorithm 15. • $k > 1$, then create a new cycle according to the MOCR heuristic of Algorithm 16.
Step 6	Add newly constructed cycle c_i to the set of cycles C and its straddling links s_i to the set of straddling links S.
Step 7	Update link weights on graph G as follows: • For every link $e \in S$, set a weight equal to ∞. • For every link $e \in C$, set a weight equal to h.
Step 8	Go back to Step 1.
Step 9	Return set of p-cycles C and their set of straddling links S.

In general, when protection and working light-trees are completely disjoint, then the IA-PMC-RWA algorithm first solves the routing and wavelength assignment problems by finding both the working light-tree (using the ST heuristic) and its backup paths (using any of the aforementioned protection techniques) on the same wavelength (on the first available wavelength from a list of wavelengths (first-fit algorithm)). The multicast request is blocked if there is no available wavelength for the entire working tree and its protection paths. If a wavelength assignment is possible, the working tree and its protection paths are combined into a working-protection tree, and the Q-factor for each path on the working-protection tree is evaluated. The multicast request is blocked if there is at least one route on that tree with a Q-value that falls below a predetermined threshold value, or there is no TX/RX available, and there is no alternate wavelength assignment possible. Otherwise, routing for a new wavelength assignment is implemented and the heuristic is repeated. Fig. 4.26 shows the flowchart for the IA-PMC-RWA algorithm that is used in this work.

Note that when self-sharing techniques are considered, many alternate paths may exist to protect the affected destinations upon a link failure and the backup paths depend on the

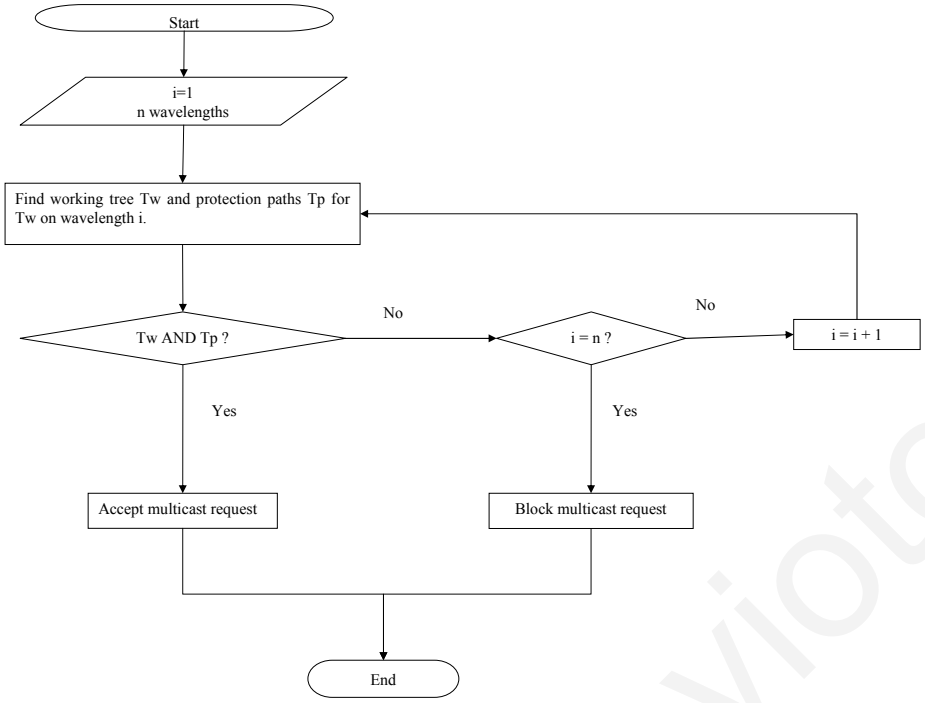

Figure 4.25: Flowchart of the protected multicast routing and wavelength assignment (PMC-RWA) algorithm.

certain link that has failed. As link failures occur in a random fashion in the network, backup paths cannot be known prior to the failure. Therefore, the problem that arises here is that once the working and backup paths are found, there is no straightforward approach to identify the worst Q-value for a destination node that is reached from the source node via its potential worst backup path. A worst-case analysis to this problem entails merging together the primary tree and its backup paths into a single graph G', and then computing in G' the longest paths from the source node to every destination node. Then the longest path to each destination can be considered as the backup path that yields the minimum Q-factor amongst every potential backup path that can be created upon a single link failure.

Thus, the maximum-cost heuristic (MXCH) was developed for this work that computes the longest paths from a specific source node in combined graph G'. The MXCH heuristic algorithm is based on Prim's algorithm that spans all the nodes in a graph, starting from the source node and adding in each iteration of the algorithm the minimum-cost link into the tree. In this case, the MXCH heuristic, instead of adding in each iteration of the algorithm the minimum-cost link, it adds to the tree the maximum-cost arc attempting to create a directed maximum-cost tree. Once the maximum-cost tree is created, the Q-factor for each destination node in the tree can be evaluated. The IA-PMC-RWA algorithm for protected

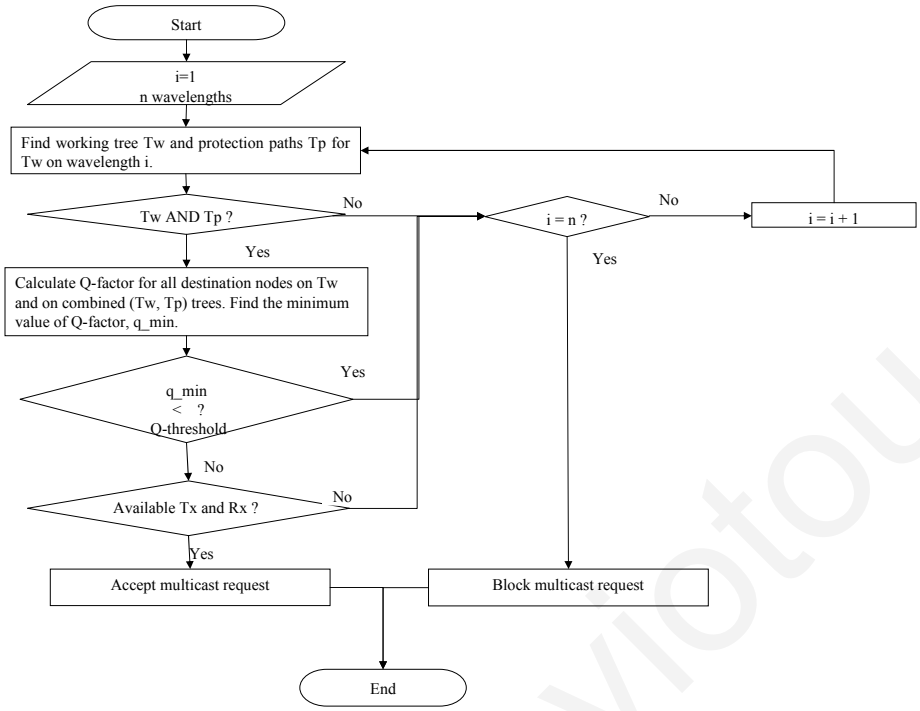

Figure 4.26: Flowchart of the impairment-aware protected multicast routing and wavelength assignment (IA-PMC-RWA) algorithm.

multicast sessions can then be modified to the following: First the routing and wavelength assignment problems are solved by finding both the working light-tree and the backup paths on the same wavelength (utilizing the first-fit wavelength assignment technique). If such routing and wavelength assignment is possible, the Q-factor for each path on the working and on the MXCH tree of G' is evaluated. The protected multicast request is blocked if there is at least one route on both working and MXCH tree of G' with a Q-value that falls below a predetermined threshold value, or there is no TX/RX available, and there is no alternate routing and wavelength assignment possible. Otherwise, routing for a new wavelength assignment is implemented and the heuristic is repeated.

4.5 Complexity Analysis

4.5.1 Segment-Based Schemes

The complexity analysis of each segment-based protection algorithm presented above is given next. For every segment-based algorithm it is assumed that n is the number of nodes in a given network, m is the number of links, while k is the number of destination nodes for a

given multicast call. Since for all three multicast protection heuristics (MCSP, MSSNF, LP) primary trees are calculated according to the Steiner Tree (ST) heuristic, the complexity of the primary tree algorithm is $O(kn^2)$ [187].

- *MCSP heuristic:* According to the MCSP heuristic, an auxiliary graph (AG) of the primary tree is created consisting only of the segment points of the tree. Then Dijkstra's algorithm is used to calculate the backup paths for each segment in AG. As pointed out, a segment is created between two segment points without passing through any other segment point in AG. Hence, if we assume that the number of segments identified in AG is s, and by considering that the complexity of Dijkstra's algorithm is $n(n+1)/2$ [51] then the complexity of MCSP is $sn(n+1)/2$. To evaluate the worst-case scenario, we have to find the maximum number of segments that can be created in an AG, connecting k destination nodes. It is found that $s = 2k - 1$ is the maximum number of segments that can be created. Thus, the complexity of MCSP heuristic becomes $(2k-1)n(n+1)/2 = (2kn^2 + 2kn - n^2 - n)/2 \in O(kn^2)$.

  Below it is shown how the worst case of $s = 2k-1$ segments is evaluated. Specifically, Fig. 4.27 illustrates the AG for different multicast group sizes, consisting of $d = 2, d = 3, d = 4$, and $d = k$ destination nodes. Destination nodes are shown with gray color, along with the root node, which is the source node of the multicast group. White nodes illustrate the branch nodes which are also segment points of the tree. In each case, segments are illustrated with an arc between two adjacent segment points of the tree. As shown in Fig. 4.27 for the general case of $d = k$, the worst-case number of segments is given by $s = 2k - 1$.

- *MSSNF heuristic:* According to the MSSNF heuristic, an auxiliary graph (AG) of the primary tree is created consisting only of the segment points of the tree. Sister nodes are identified in AG as the nodes connected to the same branch node. Segment points are all the branch nodes, destination nodes, and the source. As pointed out, $c(v)$ denotes the set of sister nodes connected to node v. If we assume that variable w_i denotes the number of nodes included in set $c(v_i)$ with $i = 1, 2, 3, ...x$ then if:

  - $w_i = 1$, then Dijkstra's algorithm is used to calculate the backup path for segment i. This however adds a complexity of $rn(n+1)/2 \in O(rn^2)$ to the algorithm where r denotes the number of $c(v_i)$ sets with $w_i = 1$.

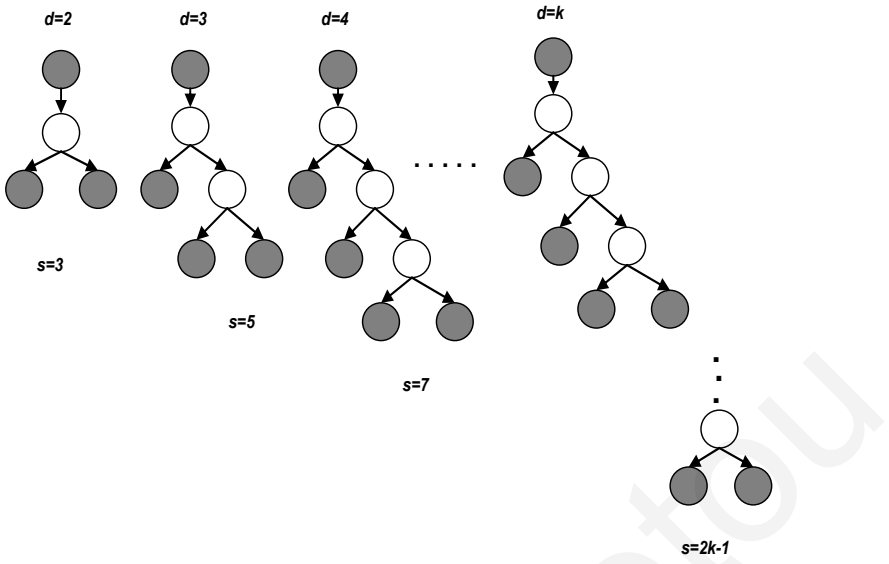

Figure 4.27: Evaluation of maximum number of segments in the MCSP scheme.

– $w_i > 1$, then the ST algorithm is used to connect nodes in set $c(v_i)$. This adds a complexity of $r'w_i n(n+1)/2$ to the algorithm, where r' denotes the number of $c(v_i)$ sets with $w_i > 1$. However, if the above fails, Dijkstra's algorithm is used to calculate the backup path for each segment between node v_i and each one of its adjacent nodes in set $c(v_i)$. Thus, the complexity that is added to the algorithm is given by $r'w_i n(n+1)/2$.

Summing up the above, the complexity of MSSNF is given by Eq. (4.1):

$$[rn(n+1) + 2r'w_i n(n+1)]/2 \tag{4.1}$$

To express the complexity of MSSNF in terms of k and n we have to identify the worst-case scenario. Again, we assume that the arrangement of nodes in Figure 4.27 corresponds to the worst case scenario, as it gives the maximum number of segments and thus the maximum number of iterations of MSSNF. According to Fig. 4.27:

– $x = k$, since the maximum number of $c(v_i)$ sets created is k.

– $r = 1$, since only $c(v_1)$ consists of only one node ($w_1 = 1$).

– $r' = 2k - 1 - k = k - 1$, since $k-1$ $c(v_i)$ sets are consisting of two nodes and thus $w_i = 2$ for $i = 2, 3, ..., k$.

By substituting variables r, r', w_i of Eq. (4.1) with the above, then the complexity of MSSNF is $[n(n+1) + 4(k-1)n(n+1)]/2 \in O(kn^2)$.

- *LP heuristic:* According to the LP heuristic, an auxiliary graph (AG) of the primary tree is created consisting only of the segmentation nodes of the tree which are defined as all the destination nodes and the source. Then the levels of the AG are identified and a backup path is created for each level segment by connecting the segmentation nodes of each level to a segmentation node that lies in any level above the current level. This is done by utilizing the ST heuristic if the number of segmentation nodes in the current level is greater than one. Otherwise, Dijkstra's algorithm is utilized. If we assume that L is the maximum number of levels in AG, and w_i is the number of segmentation nodes in level i, where $i = 1, 2, ..., L$, then the complexity of the LP algorithm is evaluated by $[w_1 w_2 n(n+1) + (w_1 + w_2)w_3 n(n+1) + ... + (w_1 + w_2 + ... + w_{(L-1)})w_L n(n+1)]/2$.

  To express the complexity of LP in terms of k and n we again have to identify the worst-case scenario. For LP, the worst-case scenario is obtained by considering the maximum number of levels that can be created by the appropriate arrangement of segmentation nodes. The maximum number of levels is evaluated as shown in Fig. 4.28.

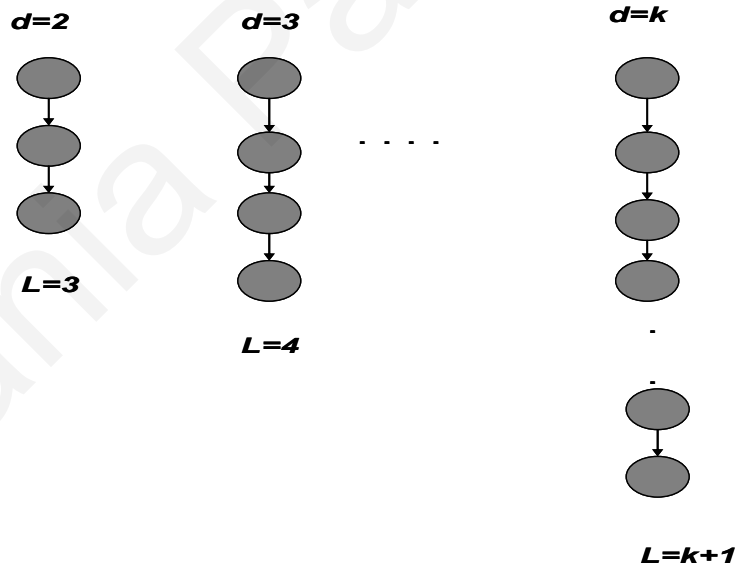

Figure 4.28: Evaluation of the maximum number of levels for the LP heuristic.

According to Fig. 4.28, $L = k - 1$ and $w_i = 1$, for $i = 1, 2, ..., L$. Hence, by substituting these numbers to the formula above, the complexity of LP becomes $[n(n+1) + 2n(n+1) + ... + (k-1+1)n(n+1)]/2 = n(n+1)(1 + 2 + ... + k) = [n(n+1)k(1+k)]/4$. Thus, the complexity of the LP algorithm is $O(k^2 n^2)$.

According to the analysis above, Table 4.12 summarizes the complexity of each segment-based heuristic. As shown in Table 4.12 the computational complexity of the LP heuristic is increased compared to the MCSP and MSSNF algorithms. Specifically, the complexity of the LP heuristic increases as the multicast group size increases. However, increased complexity of the LP heuristic does not make the heuristic impractical since a slightly longer time for the establishment of a connection might be more desirable than the blocking of a call. Note also that the computation of the protected paths is done during the provisioning of the multicast connection, prior to the occurrence of the fault. Thus, even though this algorithm is more computationally intensive, it does not affect the recovery time once a failure has occurred.

Table 4.12: Complexity of Segment-Based Heuristics

Algorithm	**Complexity**
MCSP	$O(kn^2)$
MSSNF	$O(kn^2)$
LP	$O(k^2n^2)$

4.5.2 Cycle-Based Schemes

The complexity analysis of each cycle-based protection heuristic presented above is given next. Since in cycle-based protection schemes the network is decomposed into a set of protection cycles, the complexity analysis presented here concerns the heuristics developed for the construction of these cycles. For every cycle-based heuristic, n represents the number of nodes in the network under examination and e represents the number of links in the network.

- *RC heuristic:* In each iteration of the heuristic, Dijkstra's algorithm is used for the creation of the cycles. This results to a complexity of $in(n+1)/2 \in O(in^2)$, where i represents the total number of iterations required for including each link of the network into at least one cycle. If we assume that in the worst-case scenario, during the first iteration of the algorithm only 2 links are included in the cycle created, i.e., a minimum hop cycle consists of exactly 3 links, and that in the rest of the iterations only one link is the new link added to the cycle set, then $i = 1 + (e-3) = e-2$ and the complexity of RC heuristic is $(e-2)n(n+1)/2 \in O(en^2)$.

- *PCH heuristic:* In the PCH heuristic the first cycle is created according to the OCR heuristic. If we assume that in the worst-case scenario the minimum hop cycle is created, consisting of exactly 3 nodes and links, the complexity of OCR is $n(n+1)/2$.

After the creation of the first cycle, the algorithm is repeated at most $e - 3$ times, and in each iteration one link is added to the initial cycle. The addition of new links is performed according to Dijkstra's algorithm and therefore the complexity of PCH becomes $[n(n+1) + (e-3)n(n+1)]/2 \in O(en^2)$.

- *QBPCH heuristic:* The worst case complexity of the QBPCH heuristic is given when the minimum (in hop) cycles are created. Thus, the complexity of QBPCH is $O(en^2)$. If we assume that in each iteration of the algorithm the maximum degree node has only one neighbor, then the complexity of the MOCR heuristic performed in each iteration of the QBPCH heuristic is given by Dijkstra's algorithm $n(n+1)/2$. Similar to RC heuristic, the number of iterations is found to be $e-2$. Thus, the complexity of QBPCH is $O(en^2)$.

Table 4.13 summarizes the complexity of each cycle-based heuristic.

Table 4.13: Complexity of Cycle-Based Heuristics

Algorithm	**Complexity**
RC	$O(en^2)$
PCH	$O(en^2)$
QBPCH	$O(en^2)$

4.6 Performance Results

To evaluate the performance of the different multicast protection techniques described above, a metropolitan area optical network was considered with statistics as shown in Table 4.14. Note that the topology of the metro network used in the simulations is described in Chapter 3.

Table 4.14: Network Statistics

Number of nodes	50
Number of links	98 (196 arcs)
Average node distance	60 Km
Maximum link distance	100 Km
Minimum link distance	20 Km
Average node degree	3.92
Minimum node degree	3
Maximum node degree	6
Network diameter	305 Km (6 hops).

Multicast requests arrive into the system dynamically according to a Poisson process and the holding time is exponentially distributed with a unit mean. In each simulation, $5,000$ re-

quests were generated for each multicast group size for a total of 45,000 multicast requests, and the results were averaged over five simulation runs. Sixty-four (64) wavelengths per link were utilized to evaluate the blocking probability versus the multicast group size for a network load of 100 Erlangs and the Q-threshold was set at 8.5 dBQ (BER=10^{-12}). The multicast group size of the session indicates the number of nodes participating in the multicast session (number of destinations plus 1 to account for the source). For the simulation runs, a node engineering with passive splitters and fixed TXs and RXs was assumed as shown in Fig. 3.5 in Chapter 3.

4.6.1 Segment-Based Schemes

As the LP heuristic is designed to protect directed connections, for comparison purposes, the SP and SSNF heuristics, as previously described, have been slightly modified to operate on directed light-trees. For determining the primary tree and the backup paths both self- and cross-sharing is utilized. As mentioned previously, self-sharing denotes that the different backup paths share the primary links on the primary tree while cross-sharing denotes sharing on the common backup links on the same wavelength if their corresponding primary trees are arc-disjoint. Furthermore, in the case of self-sharing, there is also a distinction between intra-self-sharing (a backup path shares bandwidth or channels on common edges with other backup paths on the backup tree) and inter-self-sharing (a backup path shares edges not only with other backup paths but also with other edges on the primary tree).

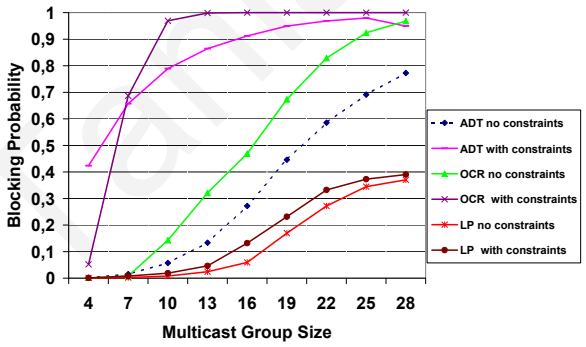

Figure 4.29: Blocking probability versus multicast group size for dedicated arc-disjoint protection techniques for impairment-unaware and impairment-aware provisioning. Results for the LP technique are also included for comparison.

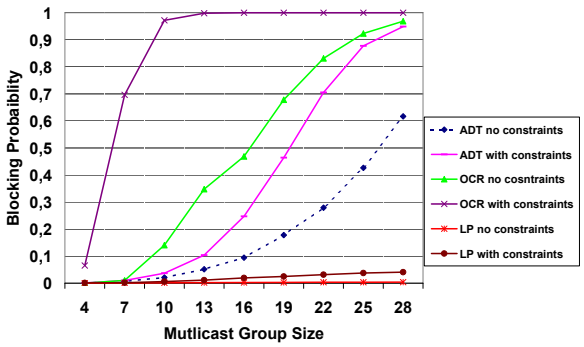

Figure 4.30: Blocking probability versus multicast group size for cross-shared arc-disjoint protection techniques for impairment-unaware and impairment-aware provisioning. Results for the LP technique are also included for comparison.

Initial results in Fig. 4.29 show the performance of the dedicated arc-disjoint protection

techniques (ADT and OCR) with and without physical layer impairment constraint considerations. For comparison purposes, the results from the LP protection technique with and without impairments are also included. The performance for dedicated arc-disjoint techniques described in this work shows that these techniques tend to create very long paths and when the physical layer constraints are taken into account, most multicast requests are blocked due to a low Q-factor leading to very high blocking probability even for small multicast group sizes. Fig. 4.30 shows the performance of the same multicast protection heuristics when cross-sharing techniques are also included. Performance results show that when cross-sharing is considered blocking probability is significantly improved. However, the blocking probability of the arc-disjoint techniques is still unacceptably high.

As the blocking probability for these techniques is unacceptably high, there are not considered any further in the performance results. Rather, various segment-based protection techniques that result in much lower blocking probability are compared. Furthermore, self-sharing techniques are also considered, since the blocking probabilities for heuristics that use sharing techniques (LP, SP, SSNF) are decreased significantly compared to those for disjoint protection schemes where there is no sharing allowed.

The rest of the results shown in Figs. 4.31– 4.34 compare the novel segment-based protection heuristic algorithm, namely LP (level protection), to the other conventional segment-based techniques, namely the MSSNF and MCSP heuristics. Specifically, Fig. 4.31 shows the blocking probability versus the multicast group size when only self-sharing is considered, while Fig. 4.32 shows the blocking probability versus the multicast group size when self-sharing and cross-sharing are jointly considered. For both of these cases, impairment-unaware provisioning was considered and it is shown that the LP heuristic algorithm performs better than existing segment-based protection approaches (in the case of self-sharing and cross-sharing the advantage is more evident for large multicast group sizes). The main reason is that the LP heuristic has more flexibility in finding the backup paths and only a single backup path is required to protect a level segment group.

Figures 4.33 and 4.34 show the blocking probability versus the multicast group size when the PLIs are also taken into account (Fig. 4.33 contains the results for when only self-sharing is considered, while Fig. 4.34 illustrates the results for when self-sharing and cross-sharing are jointly considered). In both cases, the LP heuristic algorithm significantly outperforms the conventional segment-based protection approaches. Furthermore, the results of Figs. 4.33 and 4.34 where the PLIs are considered show that the blocking probability is significantly increased when compared with the results of Figs. 4.31 and 4.32, where the PLIs were not

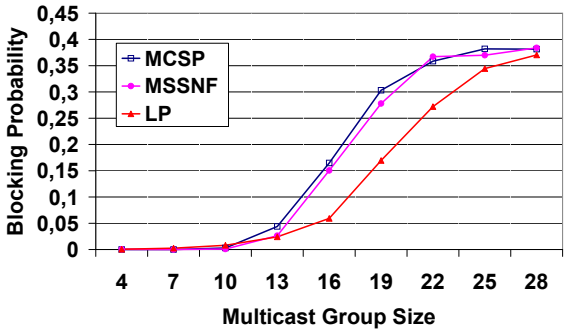

Figure 4.31: Blocking probability versus multicast group size with intra- and inter-self-sharing for impairment-unaware provisioning.

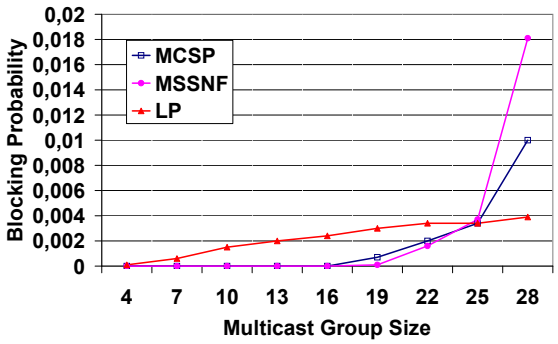

Figure 4.32: Blocking probability versus multicast group size with self- and cross-sharing for impairment-unaware provisioning.

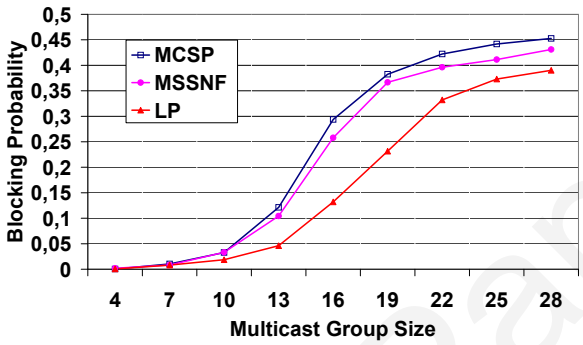

Figure 4.33: Blocking probability versus multicast group size with intra- and inter-self-sharing when PLIs are also considered.

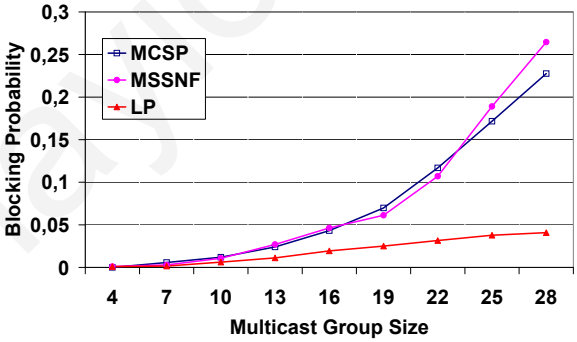

Figure 4.34: Blocking probability versus multicast group size with self- and cross-sharing when PLIs are also considered.

taken into account, an indication that more intelligent protection techniques are required that are designed in such a way so as to take into consideration the effects of PLIs. This is the case for the proposed LP heuristic algorithm that was designed aiming to reduce the length of the combined working-protection paths, thus reducing the blocking due to Q.

This is shown in Fig. 4.35, that shows the blocking probability due to Q versus the multicast group size with self- and cross-sharing when PLIs are also considered. The blocking probability due to Q for the LP heuristic is significantly lower compared to the blocking probability due to Q of the other segment-based heuristics, and so is the blocking probability due to the unavailability of wavelengths as shown in Fig. 4.36. Note that the impact of the physical layer effects increases as the multicast group size increases. Specifically, for multicast groups with more than 25 members, the blocking probability due to Q sharply increases for every multicast protection algorithm and the overall blocking probability is mainly lim-

ited by the PLIs. The drop that is observed in Fig. 4.36 for multicast groups with more than 25 members is due to the fact that for such group sizes most of the requests are blocked due insufficient Q-factor and not due to the unavailability of wavelengths.

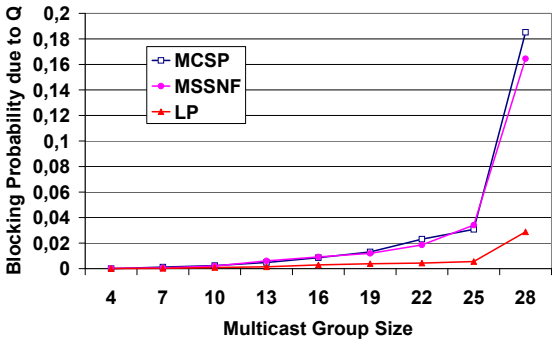

Figure 4.35: Blocking probability due to Q versus multicast group size with self- and cross-self-sharing when PLIs are also considered.

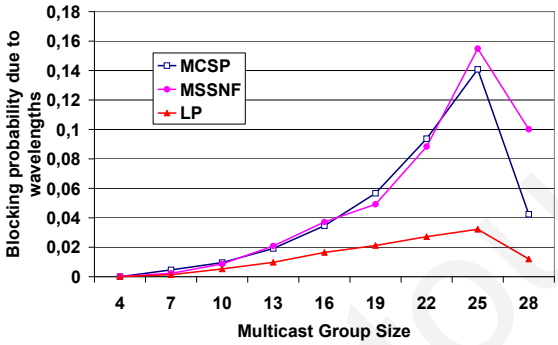

Figure 4.36: Blocking probability due to the unavailability of wavelengths versus multicast group size with self- and cross-sharing when PLIs are also considered.

The reader should note that reduced blocking probability is achieved at the expense of some increase in the computational complexity of the proposed heuristic. (The complexities of the LP, MSSNF, and MCSP heuristics are shown in Table 4.12 (where k is the multicast group size and n is the number of nodes in the network). However, this increase in complexity is not as significant in this case, as this is a protection technique and the calculation of the backup paths is done during the provisioning of the connection prior to the failure occurrence and not in real time after the failure has occurred. Thus, it is better for a multicast connection request to take some more time to be established into the network rather than being blocked while network resources are still available.

Another aspect of the protection schemes is the redundant capacity required for each heuristic algorithm for the construction of their backup paths. When self-sharing is allowed, the redundancy (R) is defined in this work by Eq. (4.2),

$$R = (1 - \frac{self - shared \quad links}{total \quad links}) \times 100 \tag{4.2}$$

where parameter *self-shared links* is defined as the number of the links shared between the backup and primary paths of the same multicast call and *total links* is defined as the number of the links used by the combined backup and primary paths.

When sharing is now allowed not only between the resources of the backup and primary paths of the same session but also between the resources of the backup paths of different

multicast calls (cross-sharing), the redundancy (XR) is now defined by Eq. (4.3) as,

$$XR = R - \frac{cross - shared \quad links}{total \quad links} \times 100 \tag{4.3}$$

where parameter *cross-shared links* is defined as the number of the links shared between the backup paths of different multicast sessions.

The reader should note that the above definitions of redundancy do not correspond to the classical definitions that can be found in the literature. In this thesis, redundancy is defined in such a way in order to examine the impact of the self- and cross-shared links for each protection heuristic. Thus, if no self- or cross-shared links exist, i.e., protection and backup paths are fully disjoint (e.g., ADT heuristic), redundancy is 100%.

Fig. 4.37 shows the redundant capacity required for a number of multicast protection schemes versus the multicast group size when only sharing between the primary and the backup paths of the same multicast call is allowed. By default, only the segment-based schemes can share capacity between their primary and backup paths and thus redundant capacity for the ADT and OCR schemes is fixed at 100% for each multicast group size. For the segment-based schemes, redundant capacity decreases as the multicast group size increases with the MCSP scheme achieving the lowest redundancy of around 73% for a multicast group of 28 members. The *LP* heuristic for the same multicast group size achieves the highest redundancy compared to the other segment-based schemes of around 93%.

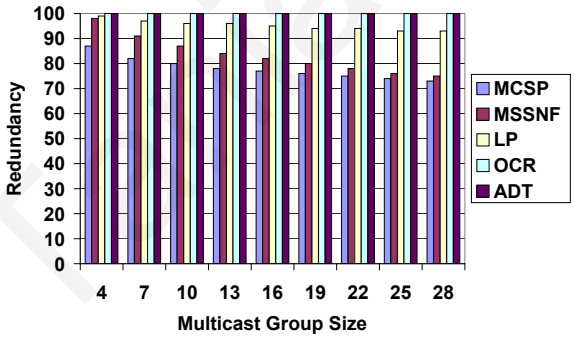

Figure 4.37: Redundant capacity required for each heuristic algorithm versus multicast group size with self-sharing for impairment-unaware provisioning.

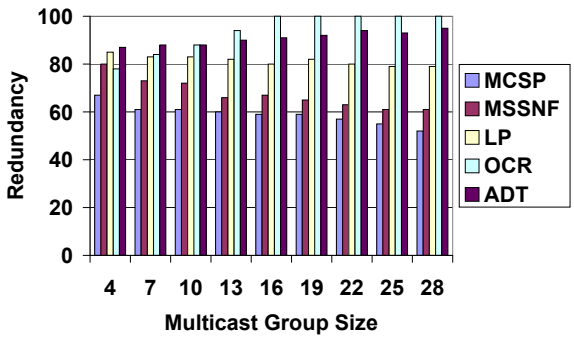

Figure 4.38: Redundant capacity required for each heuristic algorithm versus multicast group size with self- and cross-sharing for impairment-unaware provisioning.

Fig. 4.38 shows the redundant capacity required for the same set of multicast protection schemes versus the multicast group size when cross-sharing is now also allowed. Compared to Fig. 4.37, redundant capacity significantly drops for every protection heuristic while again,

among the segment-based schemes, the MCSP heuristic achieves the lowest redundancy of around 51% for a multicast group of 28 members. Again, LP achieves the highest redundancy among the segment-based schemes, which reaches 79% for a multicast session of 28 nodes. Note that for the OCR heuristic redundancy reaches 100% as group sizes increase due to the restrictive way that it builds its backup paths and hence reducing the chance of the different backup paths to share their resources.

The increased routing restrictions that the *LP* heuristic algorithm imposes compared to the other segment-based schemes discussed here, is also the reason for its increased redundant capacity. However, in terms of session blocking probability it performs better due to its flexibility in finding backup paths between successive levels. Specifically, in the LP heuristic, it is possible that more than one candidate nodes exist that can be the origin of the backup path, while in other segment-based schemes only one node can be the node from where a backup path originates. Cross-sharing gives an advantage to the other two segment-based schemes since their blocking probability is lower than LP's blocking probability, for up to a multicast group size of 25 nodes. This is due to their increased capability of cross-sharing their resources. For larger group sizes however, LP performs better due to its flexibility on finding backup paths between successive level segments.

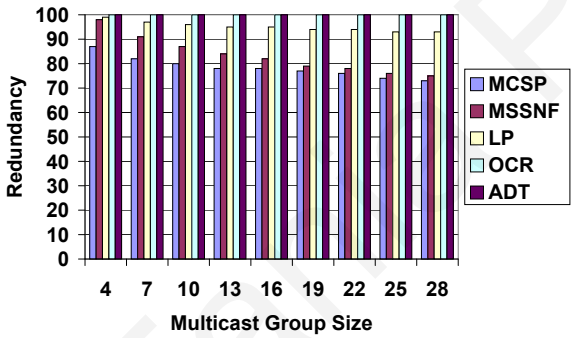

Figure 4.39: Redundant capacity required for each heuristic algorithm versus multicast group size with self-sharing when PLIs are also considered.

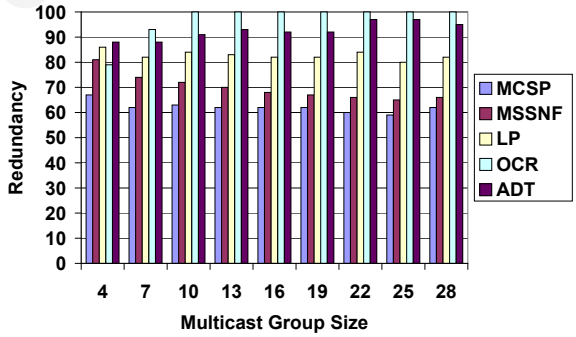

Figure 4.40: Redundant capacity required for each heuristic algorithm versus multicast group size with self- and cross-sharing when PLIs are also considered.

Redundant capacity versus the multicast group size is evaluated when the PLIs are also considered. Figs. 4.39 and 4.40 show the redundant capacity required for the set of multicast protection schemes previously discussed when only self-sharing techniques are considered and when both self- and cross-sharing techniques are considered respectively. Again, results of Fig. 4.39 are evaluated according to Eq. (4.2) while results of Fig. 4.40 are evaluated according to Eq. (4.3). Results show that in both cases redundancy is not affected by the

PLIs since results of Fig. 4.37 are similar to the results of Fig. 4.39 and results of 4.38 are similar to the results of Fig. 4.40. Once again the LP technique requires some additional redundant capacity compared to the rest of the segment-based techniques. The tradeoff here is increased redundant capacity versus lower blocking probability (due to network resources, as well as the physical layer impairments). As previously mentioned, the basic idea behind the LP (level protection) heuristic algorithm was to design a multicast protection algorithm that performs well even when the PLIs are considered (contrary to the rest of the segment-based approaches as demonstrated in Fig. 4.36). This is the case as the LP heuristic presented in this work is designed while taking into consideration the PLI constraints and by also considering the tree topology used for multicast connectivity, instead of modifying algorithms used for point-to-point connections, which is common practice in most of the literature for the multicast protection techniques.

4.6.2 Cycle-Based Schemes

Figs. 4.41 and 4.42 show the blocking probability versus the multicast group size for cycle-based protection techniques when only self-sharing is considered and when both self- and cross-sharing techniques are allowed respectively. Performance results were evaluated for the proposed PCH and QBPCH heuristics and for the existing RC and Hamiltonian Cycle (HC) schemes, developed for comparison purposes. Note that for the QBPCH approach, variable q was set at $-\infty$ that indicates that no PLIs are considered during the implementation of the QBPCH p-cycles, and thus QBPCH returns the minimum possible number of cycles that can be achieved by the heuristic. Results show that the conventional RC heuristic algorithm which finds a set of minimum (in length) cycles performs the best for both cases. The reason for this is that in the HC approach the network is decomposed into a single protection cycle and in the PCH/QBPCH heuristics the network is decomposed in fewer protection cycle(s) compared to the RC approach. Therefore, in HC, PCH, and QBPCH schemes, increased blocking probability is caused due to the unavailability of protection paths. In particular, the number of cycles constructed for each heuristic under examination are shown in Table 4.15. Note that the results of the HC approach are similar to the PCH results and that when cross-sharing is also considered (Fig. 4.42) blocking probability is significantly decreased for every cycle-based scheme.

The performance of the cycle-based heuristics is also examined when the PLIs are taken into consideration. The QBPCH heuristic, which is constructed in such a way that it takes

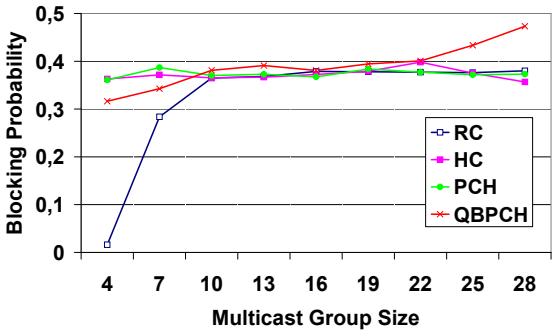

Figure 4.41: Blocking probability versus multicast group size for various cycle-based protection techniques.

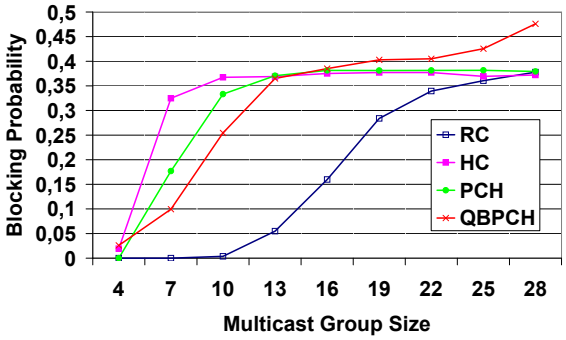

Figure 4.42: Blocking probability versus multicast group size for various cycle-based protection techniques when cross-sharing is also allowed.

Table 4.15: Cycle Information for a Network with 98 Working Links

	Number of Cycles	Number of Straddling Links
RC	29	0
HC	1	48
PCH	9	31
QBPCH, $q = -\infty$	15	17
QBPCH, $q = 8$	16	17
QBPCH, $q = 9$	20	14
QBPCH, $q = 10$	26	6
QBPCH, $q = 11$	31	2
QBPCH, $q = 12$	35	0
QBPCH, $q = \infty$	35	0

into account the PLIs during the creation of the p-cycles, is compared for different Q-thresholds q. Information about the number of cycles and the number of straddling links for each cycle-based heuristic examined is given in Table 4.15. According to Table 4.15, as Q-threshold q increases, the number of cycles increases while the number of straddling links decreases. This is reasonable since as q increases the length of the p-cycles decreases. Furthermore, Table 4.15 shows that after a certain Q-threshold ($q = 12$), QBPCH reaches its maximum number of cycles and hence has created cycles with the minimum possible length utilizing this heuristic. Note that while both RC and QBPCH for $q = \infty$ decompose the network into cycles with the minimum length, QBPCH returns 35 cycles while RC returns only 29 cycles. This is due to the fact that a different heuristic is used each time, starting the decomposition from a different node in the network.

Figs. 4.43 and 4.44 show the blocking probability versus the multicast group size when the PLIs are also taken into account. Fig. 4.43 considers only self-sharing, while Fig. 4.44 al-

lows for both self- and cross-sharing techniques to be performed. In both cases, the blocking probability for every heuristic algorithm is increased compared to the case where the PLIs were not considered. The QBPCH heuristic algorithm that sets a threshold on the length of the p-cycles, performs the best when $q = 11$ for both cases, while when cross-sharing is also allowed the blocking probability is significantly improved. In general, results of Figs. 4.43 and 4.44 indicate that by reducing appropriately the length of the cycles, the blocking probability can be improved. By reducing the length of the cycles, a p-cycle set is constructed with a larger number but shorter (in length) cycles, thus improving the Q-factor at the destination nodes. Compared to the RC heuristic, QBPCH with $q = 11$ results in a better blocking probability due to the fact that cycles in QBPCH are short enough to improve the blocking probability caused by the PLIs but not too many in number in order to increase the number of cycles required to protect a single multicast request. Furthermore, results show that in QBPCH with $q = \infty$ the blocking probability is very close to the result found for the RC heuristic, as in both cases the minimum in size and maximum in number set of cycles is attempted. Note that when the PLIs are also considered, results for both the HC and PCH heuristics are very close.

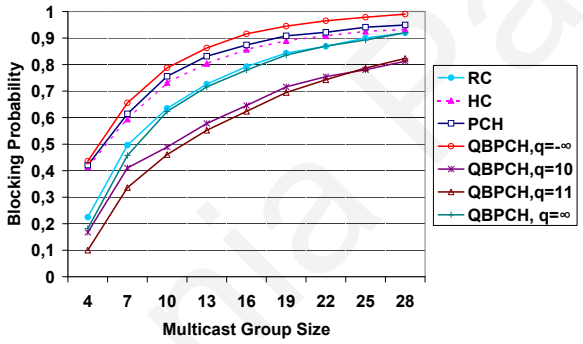

Figure 4.43: Blocking probability versus multicast group size for various cycle-based protection techniques when PLIs are also considered.

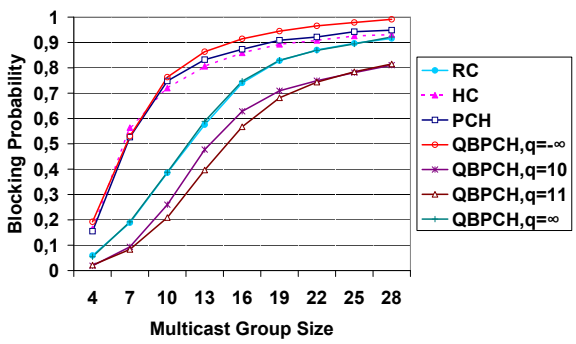

Figure 4.44: Blocking probability versus multicast group size for various cycle-based protection techniques when cross-sharing is allowed and PLIs are also considered.

In general, according to the results shown, neither the shortest nor the largest in length cycles achieve the lowest blocking probability. The lowest blocking probability is rather a trade-off between the blocking probability due to the PLIs and the blocking probability due to the unavailability of wavelengths, and for each individual network is achieved for a different Q-threshold q, depending on the characteristics of the network, the number of wavelength utilized, and the traffic load.

Apart from the blocking probability of the cycle-based schemes developed in this work,

the redundant capacity for each heuristic is also evaluated. Redundancy (R) for the case where only self-sharing is considered is evaluated according to Eq. (4.2), while for the case where both self- and cross-sharing are considered redundancy (XR) is evaluated according to Eq.(4.3).

Fig. 4.45 shows the redundancy versus the multicast group size when only self-sharing is considered and Fig. 4.46 shows the redundancy versus the multicast group size when both self- and cross-sharing are considered. Results of Fig. 4.45 show that the required redundant capacity drops with increasing multicast group size since as the multicast group size increases the number of self-shared links is increased. The lowest redundancy is achieved by the HC scheme that utilizes a single cycle for protection purposes. However, when cross-sharing is also considered (Fig. 4.46), redundancy converges to the same value as the multicast group size increases, for every cycle-based protection scheme. Initially, redundancy for small group sizes significantly improves for every scheme. As the multicast group size increases, however, the improvement resulting from cross-sharing the resources becomes insignificant and redundancy converges to the values reached previously when cross-sharing techniques were not considered.

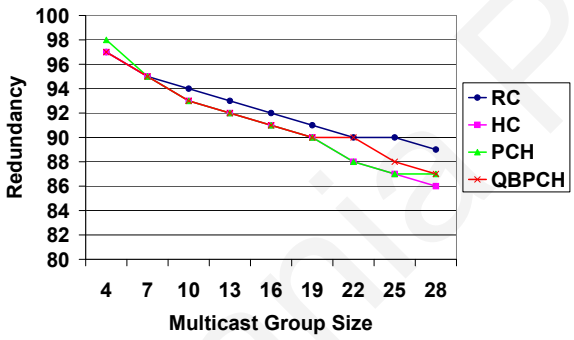

Figure 4.45: Redundant capacity versus multicast group size for various cycle-based protection techniques when only self-sharing is considered.

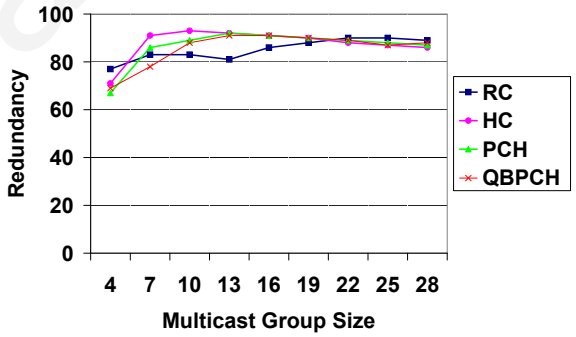

Figure 4.46: Redundant capacity versus multicast group size for various cycle-based protection techniques when both self- and cross-sharing is considered.

Redundancy is also examined for the case where the PLIs are taken into consideration. Figs. 4.47 and 4.48 show the redundancy versus the multicast group size for a number of cycle-based protection schemes when only self-sharing is considered and when both self- and cross-sharing is allowed. Results show that PLIs do not affect the redundancy of the several cycle-base schemes developed, as results of Fig. 4.45 are very close to the results of Fig. 4.47 and results of Fig. 4.46 are very close to the results of Fig. 4.48.

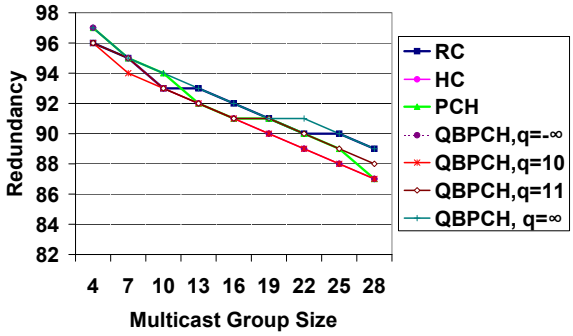

Figure 4.47: Redundant capacity versus multicast group size for various cycle-based protection techniques when only self-sharing is considered and with PLIs taken into consideration.

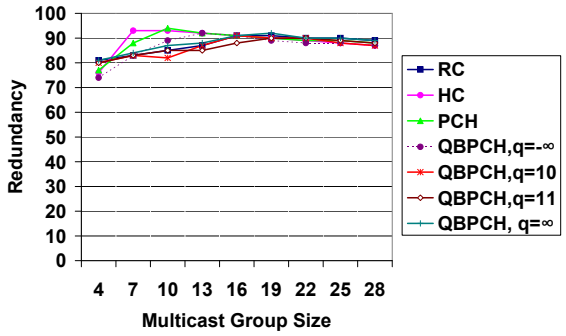

Figure 4.48: Redundant capacity versus multicast group size for various cycle-based protection techniques when both self- and cross-sharing are considered and with PLIs taken into consideration.

In general, however, based on the results of Figs. 4.41- 4.44, blocking probability is exceptional high for every cycle-based scheme developed here, even without the PLIs taken into consideration. Although constraints on the size of cycles through the use of the Q-factor can effectively reduce the blocking probability, still the blocking probability of even the best cycle-based approach is considerable high. In order to compare the performance of cycle-base schemes to other approaches like segment- and tree-based schemes, the blocking probability versus the multicast group size was also evaluated for the arc-disjoint algorithm (ADT) and the level protection algorithm (LP) previously described in this chapter. Note that simulations were performed for the same network and traffic load while the PLIs were also considered and cross-sharing was also performed. The LP heuristic is chosen for comparison purposes as it performs the best amongst the segment-based schemes previously presented while ADT was chosen as it also performs the best amongst the tree-based schemes. Results are shown in Fig. 4.49 in which QBPCH with $q = 11$ (the best cycle-based approach) is compared to the LP and ADT schemes. As shown in this figure, segment-based schemes outperform cycle- and tree-based schemes while cycle-based schemes outperform tree-based schemes only for large group sizes. However, achieving the lowest blocking probability is not always the objective for every application, as pre-configured cycles achieve ring-like high speed protection that under certain service level agreements (SLAs) could be more desirable than the high efficiency of other schemes (like segment-based schemes) for which complex protection switching mechanisms must be developed.

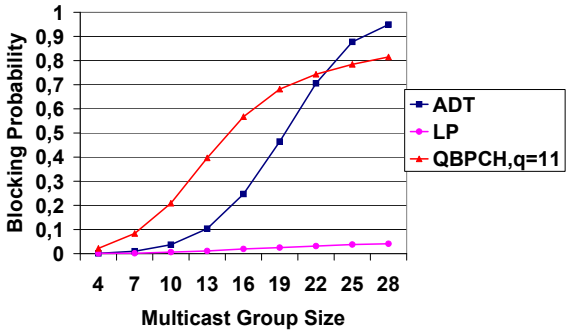

Figure 4.49: Blocking probability versus multicast group size with cross-sharing considered when the PLIs are taken into account for several protection schemes: Tree-based schemes (ADT), segment-based schemes (LP), and cycle based schemes (QBPCH, $q = 11$).

4.7 Conclusions

In wavelength routed networks it is not only important to route efficiently a connection request but also to maintain the survivability of the connection when link failures occur. Survivability is even more critical in multicast connection provisioning since multiple destinations may be affected upon a single link failure. In this chapter a novel segment-based protection heuristic algorithm, called level protection (LP), is proposed and compared to the other conventional dedicated and segment-based techniques, namely the ADT, OCR, SSNF, and SP heuristics. The different protection techniques are compared for the case when the PLIs are considered and the case when the PLIs are not taken into account. Results show that the LP heuristic performs the best, especially when cross- and self-sharing techniques are also utilized, and the PLIs are taken into consideration. This is a clear indicator that when designing protection techniques the effect of PLIs cannot be ignored for solutions that require quality-of-transmission (QoT) guarantees.

Also, two novel cycle-based protection heuristic algorithms are proposed, namely the p-cycles heuristic (PCH) and the Q-based p-cycles heuristic (QBPCH). Both heuristics are compared to a conventional cycle-based scheme, namely the ring-cover (RC) approach, and to a Hamiltonian Cycle (HC) scheme. The QBPCH approach, which sets a limit for the length of the cycles created by utilizing the constraints of the physical layer exhibited the lowest blocking probability. Specifically, the QBPCH heuristic performs better than PCH, RC, and HC schemes for every threshold that was set on the size of the cycles via the Q-factor, while the results of the PCH heuristic were very close to the results of the HC scheme. Cycle-based heuristic algorithms were also examined for the case were cross-sharing techniques were considered in which case the blocking probability was significantly improved.

Additionally, cycle-based schemes were compared to a tree-based scheme (ADT) and to a segment-based scheme (LP) and results showed that cycle-based schemes perform better than tree-based schemes only for large group sizes. Blocking probability of cycle-based schemes is significantly higher when compared to segment-based protection approaches. However, segment-based techniques require complex protection switching protocols for the signaling and switching mechanisms slowing down their protection speed compared to the high recovery speed of the cycle-based schemes.

Redundant capacity was also investigated as a performance metric and it was shown that the LP approach requires some additional redundant capacity compared to MCSP and MSSNF. The additional redundant capacity required though is a reasonable tradeoff, given the significant decrease in the blocking probability exhibited by the LP technique compared to the rest of the protection approaches.

It is noted that when sharing techniques are involved, a more complex switching mechanism is required and the link that has failed needs to be identified since different link failures lead to different backup paths. Therefore, new protection protocols need to be developed on the signaling and switching mechanisms when sharing techniques are considered.

Chapter 5

Multicast Traffic Grooming in WDM Mesh Networks

The emergence of Wavelength Division Multiplexing (WDM) technology provides the capability for increasing the bandwidth of an optical network, by grooming low-speed traffic streams onto high-speed wavelength channels. Today, a typical connection request requires bandwidth that is only a fraction of the wavelength bandwidth. Therefore, in order to efficiently utilize the capacity of each wavelength channel, several independent lower speed traffic streams (sub-wavelengths) must be multiplexed onto a single lightpath. The process of allocating sub-wavelength traffic demands to wavelength lightpaths such that the resources are shared is known as traffic grooming [83].

Most multicast service applications also require only sub-wavelength capacity. For example, HDTV requires just 20 Mbps per channel, while a normal TV channel typically requires less than 2 Mbps per channel, when compressed using MPEG-2 as in digital television. Hence, many such connections can be groomed together onto a single wavelength [192].

Traffic grooming in mesh WDM networks has received considerable attention from the research community [10, 17, 37, 48, 53, 63, 139, 153, 188, 195, 197, 221, 222, 226, 232–234]. However, in these studies, only unicast traffic was considered while only very little work has been performed on grooming multicast traffic [87, 98–102, 192, 207]. As next generation networks are expected to support both unicast and multicast applications, such as multiparty conferencing, software and video distribution and distributed computing, it is important to design and dimension networks in order to be able to support traffic of the multicast type, while grooming sub-wavelength traffic demands. Furthermore, as network architec-

tures change from ring-based to mesh-based, both unicast and multicast traffic grooming in mesh-based networks will become an important extension to current ring-based grooming algorithms.

This thesis addresses the problem of dynamically provisioning low-speed multicast connection requests in wavelength division multiplexed (WDM) optical networks with arbitrary mesh topologies. Specifically, it focuses on building a dynamic logical topology where light-trees are set up and torn down in response to dynamic multicast traffic demands. For dynamic traffic grooming at the network operation stage, where some resources have already been deployed in the network and will remain unchanged for some time, the objective is to maximize the network throughput or minimize the blocking probability of the connection requests. To achieve this objective, the grooming algorithm must utilize the overall network resources as efficiently as possible by selecting the most appropriate combination(s) of the available physical and logical resources.

The proposed approach focuses on building a dynamic hybrid topology consisting of both the available physical and logical resources and then routing the arriving multicast call on the hybrid graph. For the construction of the hybrid graph several schemes are proposed for solving the problem that arises when more than one logical links are created between two nodes (this problem is related to the bin-packing optimization problem). Specifically, each scheme is based on a different characteristic of the available logical links; this characteristic can be either their free capacity, the number of calls they are already serving, or the matching of their source/destination nodes to the new multicast request. For example, if their free capacity is considered, the maximum free capacity or the minimum free capacity link is chosen to be placed first on the hybrid topology. By doing so, the multicast traffic routing and grooming problems are solved jointly as the available resources can be explored over a single (hybrid) graph topology. The proposed approach that routes/grooms the traffic on hybrid graphs is compared to existing approaches that explore available physical and logical resources separately and sequentially. Attempts for solving the traffic grooming problem jointly have also been made in [232, 233]. In these works however, only unicast connection requests were assumed. In addition, in [232] full wavelength conversion at every node is present and the problem of multiple links between two nodes is not considered.

Furthermore, this chapter addresses the problem of traffic grooming of multicast subrate connections in conjunction with physical layer impairments. Impairment-aware grooming techniques for multicast connections are investigated for the first time in this thesis and will serve as the reference point for future works on multicast traffic grooming in optical networks

where OEO conversions occur only at the grooming part of the nodes while the rest of the pass-through traffic remains in the optical domain throughout.

5.1 Traffic Grooming State-of-the-art

Similar to unicast traffic grooming, the problem of multicast routing/grooming and wavelength assignment is also NP-complete [20] and therefore effective heuristic algorithms need to be proposed to address it even in the absence of physical layer impairments. The problem of unicast traffic grooming in WDM ring and mesh optical networks is extensively studied in several survey papers such as [55, 83, 119, 235] and it will not be reiterated here as the focus of this thesis is for multipoint connections.

The problem of grooming multicast traffic in optical networks is an important problem that has received little attention given its practical importance [43, 87, 192, 207]. Authors in [192] proposed an integer linear programming (ILP) formulation in order to minimize the number of wavelength channels used and the cost of the network in terms of the number of SONET add/drop multiplexers (ADMs). In that work, the network was represented as three different levels, namely the physical, the lightpath, and the connection levels. The authors considered nonuniform static traffic and they also introduced heuristics to solve the problem by obtaining first an initial solution using a shortest paths tree (SPT) routing heuristic and the first-fit wavelength assignment technique, and then iteratively improving it by exploring other routes. Furthermore, authors in [207] also formulated an optimization problem for the design of a light-tree-based logical topology. That problem consisted of two sub-problems, namely the MC-RWA, and the design of a light-tree-based logical topology for multicast streams. In that work, ILP formulation was used for the design of optimum light-trees and then the light-tree-based logical topology was modeled as a hypergraph over which static multicast streams were routed.

Work in [43] proposed the Maximizing Minimum Freeload Algorithm (MMFA), in which a session provisioning strategy for dynamic multicast traffic was introduced. The MMFA algorithm aims at increasing the resource utilization in order to minimize the blocking probability for future arriving requests. This was done by creating for each arriving request W "freeload" graphs, one for each wavelength in the network. The freeload graphs were created based on the free capacity of each wavelength at that time and were used during the routing algorithm. In particular W multicast trees were obtained using the shortest paths trees (SPT) heuristic, and the multicast tree that yielded the maximum freeload among the W trees was

chosen. In this approach, wavelength conversion was present in the network thus eliminating the wavelength continuity constraint.

Finally, in [87] the problem of dynamically provisioning both low-speed unicast and multicast connection requests in mesh-based WDM optical networks was investigated. In that work, several routing/provisioning schemes to dynamically provision unicast and multicast connection requests were presented. In addition, a constraint-based grooming strategy was devised to utilize the overall network resources as efficiently as possible, and based on this strategy several different sequential multicast grooming heuristics were presented.

In most of the works mentioned above, grooming was performed by considering separately the logical and physical layers. Attempts for solving the traffic grooming problem jointly have been made in [232, 233] and additional related references. In these works, however, only unicast connection requests were assumed. Furthermore, in [232] full wavelength conversion at every node is present and the problem of multiple links between two nodes is not considered. The work in this thesis assumes that hybrid graphs are created by the combination of both physical and logical layers and therefore for comparison purposes heuristics that correspond to the approach where physical and logical layers are treated separately have also been developed. Specifically, two multicast traffic grooming heuristics, namely the logical first hybrid routing (LFHR) and the physical first sequential routing (PFSR) approaches, proposed in [87] were developed and compared to the hybrid routing schemes proposed here. In Section 5.3 the existing LFHR and PFSR heuristics, adapted and modified to account for the PLIs, are briefly explained.

5.2 Node Architecture for Multicast Traffic Grooming

A grooming-capable node, must be able to switch and pack lower-speed traffic streams into higher-speed streams. In general, nodes in an optical network are equipped with optical add/drop multiplexers (OADMs), or optical cross-connects (OXCs). A lightpath carrying traffic can be terminated at a node, using line terminating equipment (LTE), either for further processing and multiplexing, or for delivery purposes. Upon termination, the lightpath undergoes optical-to-electrical (OE) conversion and it is then processed by the LTE. The LTE can add or drop low-rate traffic tributaries from the aggregate channel stream and then it can send the wavelength back to the OADM/OXC in optical form (electrical-to-optical (EO) conversion). The wavelength, can then be multiplexed with other wavelengths in the outgoing fiber. As the cost of the higher-layer electronic processing equipment (LTE), dom-

inates over the cost of the optical equipment (transceivers) in WDM networks, some of the research up-till-now has focused on reducing the total number of the electronic components in the network while accommodating a given traffic demand. However, in current networks, nodes with grooming capabilities have dominated over nodes that cannot groom subrate connections onto high-rate connections (especially in the core network arena). Thus, the assumption in this work is that all the network nodes in the networks investigated will be grooming-capable.

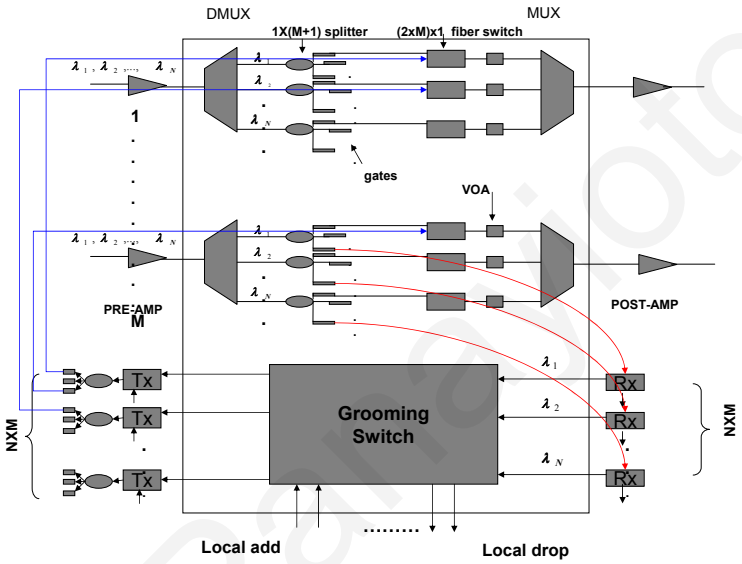

Figure 5.1: Grooming- and multicast-capable node architecture.

In order to support multicast traffic, data must be copied and duplicated using special electronic hardware (opaque node architectures) or the optical signal may split at a node to several outgoing fibers (transparent node architectures), or a combination of both (translucent node architectures). As the cost of passive optical splitters is considerably less than the cost of electronic LTE, it is more cost-efficient to duplicate the incoming traffic in the optical domain. However, electronic devices are still needed if traffic needs to be added (groomed) to wavelength channels. Therefore, in this thesis work the translucent node architecture shown in Fig. 5.1 is assumed that implements both electronic duplication and optical light splitting based on need and cost. This architecture is an extension of the node architecture with passive optical splitters and fixed TXs/RXs shown in Fig. 3.5, in which a traffic grooming fabric (GF) is added to support multicast traffic grooming.

The architecture of Fig. 5.1 can provide all-optical bypass for light-trees passing through the node without any electronic processing. Alternatively, a light-tree can be dropped if

the node is the final destination of all the traffic carried by the light-tree or otherwise the light-tree is dropped to the GF where other traffic can be multiplexed (groomed) onto this light-tree to increase the bandwidth efficiency of wavelength channels. Upon grooming, the traffic is switched to a corresponding optical splitter to split the incoming signal to multiple output ports. Note that this allows for traffic to be forwarded from one light-tree to another until it reaches its ultimate destinations. This is referred to as *multicast multihop grooming*. Note that according to Fig. 5.1, grooming is not possible at destination nodes that are intermediated nodes for a light-tree, as at these nodes the signal is dropped but not retransmitted. A portion of the signal is dropped to the destination nodes, and the rest passes through the node transparently to reach its next destination. However, the signal at the destination nodes can be dropped to the grooming switch, if the signal carries traffic for different connections, in order to allow for part of the traffic to be dropped locally and to allow for the rest of the traffic to be groomed to another light-tree if needed.

The node architecture of Fig. 5.1 is engineering for a +5 dBm optical power launched into the system. Each node's EDFA is assigned a typical noise figure which depends on its gain (Table 3.2). The gain of each pre-amplifier compensates the loss of each preceding fiber span with a fiber loss of 0.3 dB/Km and both pre- and post-amplifiers are engineered for bringing the power back to the value of +7 dBm. Variable optical attenuators (VOAs) are responsible for attenuating the total power to a prescribed value when needed, since the node design includes passive optical splitters. Specifically, in this design the prescribed value is set at -6.1 dB which corresponds to the worst-case scenario. The worst-case scenario is defined here as the maximum insertion loss caused when the signal passes through the maximum degree node. Thus, according to Table 3.1 and the fact that the maximum node degree of the network used for the simulations is 6, a 13.6 dB insertion loss is assumed for the add channels and a 16.6 dB loss for the pass-through channels at each node. Note that for the drop channels the insertion loss depends on the actual fanout of the destination node, as signals are dropped to the destination nodes before facing the VOA attenuation. Insertion loss is calculated based on the worst-case scenario, considering passive splitters, and the amplifier gain is set for the worst-case scenario as well. Controllable semiconductor optical amplifiers (SOAs) are also introduced as gates to block the power at outputs where the signal is not destined for. All gates are controlled together in an intelligent manner to avoid clashing at the same output port and/or same wavelength of the switch. At the destination nodes, p-i-n photodiodes are used and RX pre-amplifiers have noise figure of 4.5 dB. The number of TXs/RXs is assumed to be equal to the number of wavelengths times the degree of the node.

5.3 Existing Multicast Traffic Grooming Schemes

The authors in [87] investigated the dynamic multicast traffic grooming problem for which two heuristic approaches, namely the Logical First Hybrid Routing (LFHR) and the Physical First Sequential Routing (PFSR) heuristics were proposed amongst others. During both heuristics, the physical and logical layers of the network are treated separately and sequentially. Specifically, upon the arrival of a multicast call, the PFSR heuristic first seeks a new light-tree from the source to the destination nodes and only if this fails, the existing (single or multihop) logical routes are then inspected. For example, in Fig. 5.2 where logical and physical layers are shown separately, two requests are already established into the network when a new multicast connection from source node A to destination nodes F, C is accepted. It is assumed that this network operates utilizing one wavelength only. Based on the description above, PFSR first routes the request on the physical layer from node A to node F. Then, since destination node C cannot be reached from the physical layer (no available wavelength), free capacity on logical link (F, C) is used to connect node C to the source node of the request.

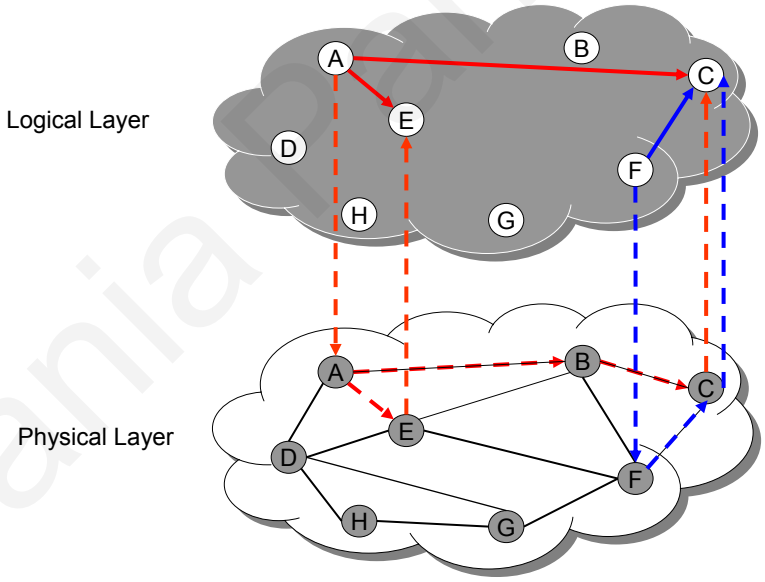

Figure 5.2: Example of the PFSR and LFHR grooming heuristics.

Unlike PFSR, LFHR first seeks existing light-tree(s) to provision the request or part of the source/destination nodes in the request, and if this fails or only part of the request is served, a new light-tree is created. In this case, for the example of Fig. 5.2, the new multicast call with source A and destination nodes F and C is first provisioned on the logical layer by reusing logical link (F, C) and then the physical layer is examined for a lightpath that will

connect the source node of the request to the link (F, C). Based on Fig. 5.2, there exists an available lightpath $(A - D - H - G - F)$ from node A to node F.

As pointed out, both heuristics are decomposed into the physical provisioning phase and the logical provisioning phase and depending on the heuristic followed, the two phases are performed sequentially. The LFHR and PFSR heuristics were implemented in this work for comparison purposes as described below. Note that while the basic idea of the LFHR and PFSR heuristics is used for sequential routing, new heuristics are developed during the logical and provisioning phases in order for the LFHR/PFSR approaches to be comparable to the hybrid approached proposed in this work. In particular, for the logical provisioning phase, the maximum overlapping light-tree (MOL) heuristic is developed as described next, while for the physical provisioning phase, the multicast routing and wavelength assignment (MC-RWA) algorithm is extended to account for the PLIs and the TXs/RXs constraint. Also, the Steiner Tree (ST) heuristic for routing multicast calls, is extended to the sub-Steiner tree (SST) heuristic to allow for feasible sub-light-trees to be established into the network.

5.3.1 Logical Provisioning

For the logical provisioning phase, the maximum overlapping light-tree (MOL) heuristic is developed aiming at minimizing the session blocking probability by grooming each multicast call onto the maximum overlapping light-tree. Each time a multicast call arrives into the network, MOL computes a list L consisting of the already established light-trees with free capacity that is equal to or greater than the rate of the new call. Then, each light-tree in L is compared to the new multicast call, and the light-tree that can serve the maximum number of source/destination nodes is chosen for grooming the new call. The basic steps of the MOL heuristic for a multicast request with rate r and multicast set $R = [s, D]$, where s stands for the source node and D is the destination set, are given in Table 5.1. Note that in Table 5.1 parameter h is also included as a constraint to the maximum number of logical hops MOL is allowed to perform.

According to Table 5.1, if the MOL heuristic is used in conjunction with the PFSR approach, a multicast request is blocked if there are no logical light-tree(s) that can accommodate the set of nodes that the physical provisioning phase failed to accommodate; otherwise the multicast request is accepted. However, if MOL is used in conjunction with the LFHR approach, a multicast request is accepted if it can be groomed onto one or more light-trees. Otherwise, if only a subset of the multicast call can be accommodated in the logical layer

Table 5.1: Basic Steps of the MOL Heuristic

Step 1	Initialize $w = 1$.
Step 2	Create list L consisting of all the currently established light-trees with free capacity equal to or greater than r.
Step 3	For each light-tree i in list L count the number k_i of source/destination nodes that are similar to nodes in set R, and associate each light-tree to its k_i number.
Step 4	Identify light-tree $R' = [s', D']$, with its k_i number being the maximum amongst the other light-trees in list L.
Step 5	If $k_i > 0$ then go to Step 6 else go to Step 9.
Step 6	Multiplex R with R' and according to the overlapping case that occurs update multicast set R as follows: • If both R and R' have the same source and also every destination node in set D is included in set D', then remove every element included in set R, in such a way that R becomes empty. (If $s = s'$ and $(D \cap D') = D$, then $R = \emptyset$). • If R and R' do not have the same source but every destination node in set D is also included in set D', then $R = [s, s']$. (If $s \neq s'$ and $(D \cap D') = D$ then $R = [s, s']$). • If both R and R' have the same source but only some of the destination nodes in D are included in D', with D_r representing the set of destination nodes included in D but not in D', then $R = [s, D_r]$. (If $s = s'$ and $(D \cap D') \subset D = D_s$ then $R = [s, D - D_s]$). • If R and R' do not have the same source and only some of the destination nodes in D are included in D', with D_r representing the set of destination nodes included in D but not in D', then $R = [s, s', D_r]$. (If $s \neq s'$ and $(D \cap D') \subset D = D_s$ then $R = [s, s', D - D_s]$).
Step 7	Increase w by one and if $w > h$ then go to Step 9, else go to Step 8.
Step 8	If $R \neq \emptyset$ then go to Step 2, else go to Step 9.
Step 9	Return R.

(MOL heuristic returns $R \neq \emptyset$), the physical provisioning phase takes place as described in the section that follows.

Figs. 5.3- 5.6 are used as an illustrative example of the MOL heuristic. In this example, it is assumed that only one wavelength is present utilizing 4 units of capacity, while every arriving request requires only 1 unit of capacity. For simplicity, the PLIs and the availability of TXs/RXs are not considered in this example. Parameter h is set to value ∞, allowing for an unlimited number of logical hops.

Fig. 5.3 shows that two light-trees, R'_1 and R'_2, are already established into the network when multicast request R_1 arrives. Since only light-tree R'_2 overlaps with the R_1 request,

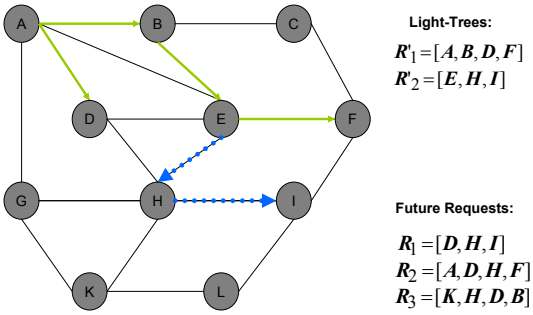

Figure 5.3: MOL example: Current network state upon the arrival of multicast call R_1,

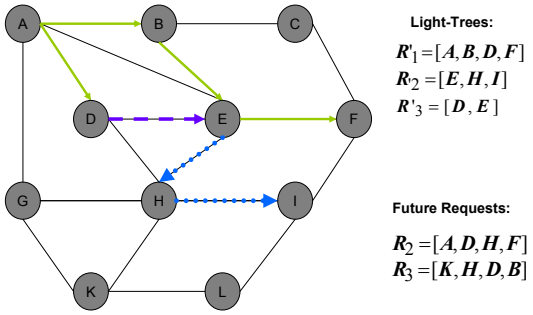

Figure 5.4: MOL example: Current network state upon the arrival of multicast call R_2.

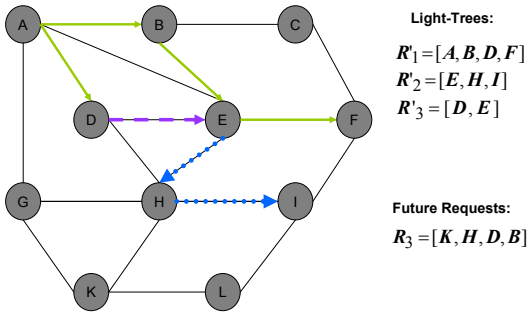

Figure 5.5: MOL example: Current network state upon the arrival of multicast call R_3

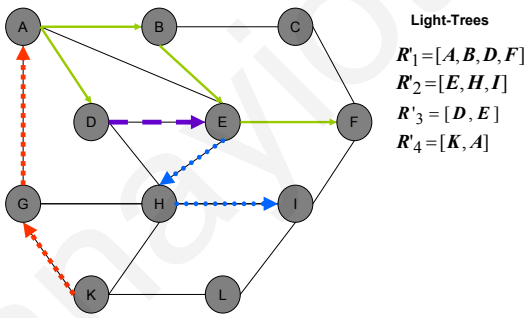

Figure 5.6: MOL example: Network state after the establishment of request R_3.

in the sense that both have the same set of destination nodes, MOL multiplexes the R_1 call with R'_2. Light-tree R'_2, however, originates at source node E, while R_1 requests a connection from source node D. Therefore, a new lightpath needs to be established to serve the updated multicast set $R_1 = [D, E]$. The updated multicast set is computed by removing from R_1 the destination nodes served by R'_1 and by adding the source node of R'_1 to the destination set of R_1. Since no logical lightpaths overlap with the updated R_1, a new lightpath is established between nodes D and E. Thus, a two-hop hybrid path is computed to serve the R_1 call.

Fig. 5.4 shows the state of the network upon the arrival of multicast call R_2. According to the current network state, two light-trees, R'_1 and R'_2, overlap with the new call. However, light-tree R'_1 has two similar destination nodes compared to R_2 while R'_2 has only one. Therefore, R_2 is multiplexed with R'_1 and since only a subset of the destination set in the multicast set is served by R'_1, R_2 is updated to $R_2 = [A, H]$. The updated multicast set overlaps with R'_2 since both have the same set of destination nodes. Thus, R_2 is again updated to $R_2 = [A, E]$ that overlaps with R'_3, and R_2 is then updated to $R_2 = [A, D]$. As destination node D is

already reached by the R'_1 light-tree, no additional logical hop is required.

Similarly, when R_3 arrives into the network (Fig. 5.5), the MOL heuristic grooms R_3 onto R'_1 that serves a subset of the destinations in R_3. Then R_3 is updated to $R_3 = [K, H, A]$ that overlaps only with R'_2. Thus, multicast set R_3 is then updated to $R_3 = [K, A]$ since destination node D is already reached by R'_1. Since R_3 does not overlap with any other established light-tree, a physical lightpath is searched for to connect node A with node K, as shown in Fig. 5.6.

5.3.2 Physical Layer Provisioning

During the *Impairment-Unaware* physical provisioning phase, the Multicast Routing and Wavelength Assignment (MC-RWA) is solved for a multicast set R. The MC-RWA algorithm, first solves the routing sub-problem and then assigns wavelength for that route based on the first-fit algorithm. The entire multicast set is accepted if a route and a wavelength assignment is possible for multicast set R.

During the *Impairment-Aware* physical provisioning phase, the Impairment-Aware Multicast Routing and Wavelength Assignment (IA-MC-RWA) must be solved. The IA-MC-RWA algorithm developed, first solves the routing sub-problem and then assigns a wavelength for that route based on the first-fit algorithm. The entire multicast set is accepted, if:

- A route and a wavelength assignment is possible for the multicast set.
- The Q-factor for each path on the multicast tree is above the predetermined Q-threshold.
- There are available TXs/RXs for that connection.

If the physical impairment constraints are not met, a new wavelength assignment is implemented and the heuristic is repeated until no new wavelength assignment is possible.

Note that for the LFHR approach the Steiner Tree (ST) heuristic algorithm, described in Algorithm 1, is used for the routing sub-problem. However, for the PFSR approach the ST heuristic is slightly modified to the sub-ST (SST) heuristic in order to permit for feasible sub-light-trees to be established (a sub-light-tree contains the source node and part of the destination set). Specifically, in case the entire light-tree is not feasible, the members of the multicast set that were not able to be accommodated by the physical layer can now be accommodated by the logical layer. In particular, the SST heuristic examines the light-tree constructed by the ST heuristic on a particular wavelength, and removes from the tree

the destination node(s) with a Q-factor that is below the predetermined Q-threshold and the destination node(s) for which there are no available receivers. By the end of the SST heuristic a feasible sub-light-tree is created while the remaining destination nodes are forwarded to the logical provisioning phase. The SST heuristic is described in Algorithm 17. Note that the SST returns a sub-light-tree T and a set D_r consisting of the destination nodes not reached by T. What is actually forwarded for routing to the logical layer is the multicast set $R = [s, D_r]$ where s is the source node of tree T. However, if $D_r = \emptyset$ means that the multicast set can be fully accommodated by the physical layer and thus the logical provisioning phase is not invoked.

5.4 Routing/Grooming on Hybrid Graphs

A novel hybrid routing algorithm is proposed aiming at improving the session blocking probability by performing routing/grooming on pre-calculated *HGs* consisting of both the available physical links and logical light-trees with free capacity. A *HG* is created for each wavelength in the network upon a multicast connection arrival, based on the available physical links and the logical light-trees with free capacity at the time. Once the *HGs* are created, an extension of the Steiner tree heuristic, namely the Hybrid Steiner Tree (HST) heuristic, is used to calculate the hybrid light-tree that may be consisting of both physical and logical links.

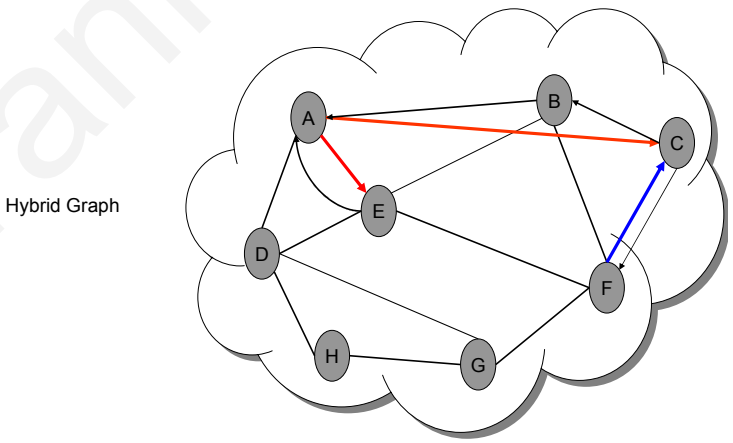

Figure 5.7: Hybrid graph created based on the physical and logical layer graphs.

Fig. 5.7 is used as an illustrative example of a *HG* created by the combination of logical

Algorithm 17 Sub-Steiner Tree (SST) Heuristic

Input: A graph $G = (V, E)$ representing the network, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$, link weights representing the physical distance assigned to each edge $e \in E$ and a Q-threshold q.

Output: A tree T spanning the set $[s, D']$ where $D' \subseteq D$, and the set $D_r \subseteq D$, where $D_r = D - D'$.

```
1: begin
2: k ← 0
3: D_r ← ∅
4: Find set D'' ⊆ D, of the destination nodes for which receivers are not available in the
   network. Let r be the number of destination nodes in D''.
5: D = D − D'', n = n − r and D_r = D_r + D''.
6: if n > 0 then
7:    T ← s
8:    while k < n do
9:       Calculate all shortest paths from nodes ∈ T to destination nodes ∈ D
10:      if At least one shortest path can be created from nodes ∈ T to a destination node
         ∈ D then
11:         Choose the shortest path amongst them and add it to tree T.
12:         Identify node d_j ∈ D last added to tree T
13:         Calculate Q-factor q_j of d_j.
14:         if q_j < q then
15:            Remove shortest path last added to T.
16:            Remove destination node d_j from D.
17:            Add destination node d_j to D_r.
18:            k ← k + 1
19:         else
20:            Remove destination node d_j from D
21:            k ← k + 1
22:         end if
23:      else
24:         D_r = D_r + D
25:      end if
26:   end while
27: end if
28: return T and D_r
```

and physical layers shown in Fig. 5.2. In this example it is shown that the two light-trees already established in the network, do not overlap in the logical layer, which makes the combination of the two layers trivial. In other systems however, more than one logical links may be present, originating from the same source and reaching the same destination node(s). For example, if lightpath $A-D-E$ was also present in the physical topology of Fig. 5.2, with source node A and destination node E, then two logical links (A, D) would be present in the logical topology of Fig. 5.2. The problem that arises here concerns the creation of the HG in which case only one logical link must be chosen for the connection of any two nodes in the

HG. This is done in order to save resources and eventually reduce the blocking probability in the network. Thus, several schemes are developed for building the HG, namely the minimum free capacity light-tree first (MCF) and the maximum free capacity light-tree first (MXCF) scheme that prioritize logical links according to their free capacity, the least-used light-tree first (LUF) and the most-used light-tree first (MUF) schemes that prioritize logical links according to the number of times that they are already being used by multicast calls already established into the network, and the maximum overlapping light-tree first (MXOF) scheme that prioritizes logical links according to their number of similar source/destination nodes(s) to the new multicast call. This is similar to the bin-packing optimization problem that appears in the literature [76] when a packet must be packet in one of several available bins that have different available capacity/space. In the next section the MCF, MXCF, MUF, LUF, and MXOF schemes developed for the construction of the *HG* are described in detail.

5.4.1 Building the Hybrid Graph

As pointed out, the *HG* is created upon the arrival of a multicast request, based on both the available physical links and the light-trees that have already been created at the time of the new request. More precisely, given a network *G*, hybrid graph *HG* is created on wavelength λ_i by first adding to *HG* all nodes of graph *G* and then adding to *HG* all the arcs with full wavelength capacity on wavelength λ_i. Then the light-trees already established in the network are examined and the light-trees with free capacity less than the rate of the arriving request are discarded. The rest of the light-trees are placed in a sorted list which is created according to one of the schemes described below.

- *Minimum Free Capacity Light-Tree First (MCF)*: The light-tree with minimum free capacity is placed first on the list attempting at loading the light-trees with less free capacity first (max reuse/pack approach).

- *Maximum Free Capacity Light-Tree First (MXCF)*: The light-tree with maximum free capacity is placed first on the list, attempting to uniformly spread the traffic.

- *Least-Used Light-Tree First (LUF)*: The light-tree that serves the minimum number of calls already established into the network is placed first on the list.

- *Most-Used Light-Tree First (MUF)*: The light-tree that serves the maximum number of calls already established into the network is placed first on the list. MUF aims at reducing the blocking probability caused due to the unavailability of TXs/RXs.

- *Maximum Overlapping Light-Tree First (MXOF)*: The light-tree which has the maximum number of similar source/destination nodes with the new multicast call is placed first on the list, attempting to decrease the number of logical hops required to serve every destination node in the request.

Note that in every scheme, the light-tree with the minimum free capacity is placed first in the list in case of a tie (i.e., more than one logical links are used by the same number of requests in the LUF or MUF schemes).

Figs. 5.8 and 5.9 are used as an illustrative example of the MXOF scheme. Fig. 5.8 shows the state of the network G upon the arrival of multicast call $R = \{a, b, h\}$ requesting 2 units of capacity and Fig. 5.9 shows information regarding the wavelength, the free capacity, the overlapping degree, the number of calls established for each light-tree and the logical links for each multicast call already established in the network. The logical links of a light-tree are defined here as all the links created by connecting the source node of the tree to each destination node. Note that a logical link between two destination nodes is not possible according to the above definition, since that would simply imply that grooming is possible at destination nodes that are intermediated nodes to the tree. Grooming at these nodes however, is not possible since, as mentioned above, although the signal is dropped at these nodes, it is not retransmitted. A portion of the signal is dropped to the destination node, and the rest passes through the node to reach its next destination. In the example of Fig. 5.9, as the arriving multicast request requires 2 units of capacity, none of the already established light-trees is discarded and based on the MXOF scheme, the light-trees of Fig. 5.9 are placed in the table in a decreasing order according to their overlapping degree. Note that the same example can be used for the MCF scheme as well, since light-trees of Fig. 5.9 are placed in the table in an increasing order according to their free capacity.

The HG for wavelength λ_i is created by first copying G to HG and then removing from HG all the physical links that are occupied by λ_i (Fig. 5.10). Subsequently, the logical links for each light-tree shown in Fig. 5.9 are added sequentially in the HG. During this procedure, if two nodes are found to be already connected in HG, either via a logical or via a physical link, then every logical link included in the conflicting light-tree is neglected. For example, Fig. 5.9 shows that λ_i and λ_w have a common link. However, since logical link (a, f) of λ_i is added first on HG, then none of the logical links of λ_w are included in HG. The resulting HG is shown in Fig. 5.11.

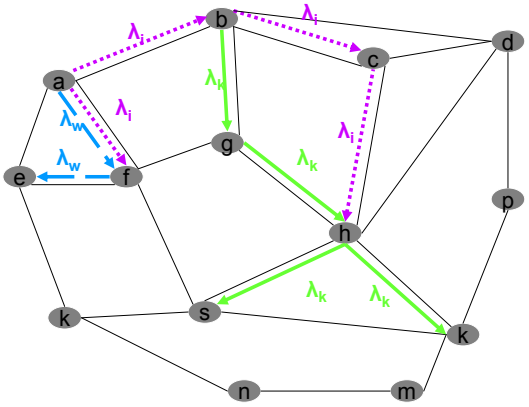

Figure 5.8: Current network state.

Multicast Group	Lambda	Free Capacity	Overlap. Degree	Number of Calls Already Serving	Logical Links
{a,b,f,h}	λ_i	3	3	1	(a, b) (a, f) (a, h)
{b,h,s,k}	λ_k	4	2	1	(b, h) (b, s) (b, k)
{a,f,e}	λ_w	6	1	1	(a, f) (a, e)

Figure 5.9: Information for already established light-trees upon the arrival of multicast call $R = \{a, b, h\}$.

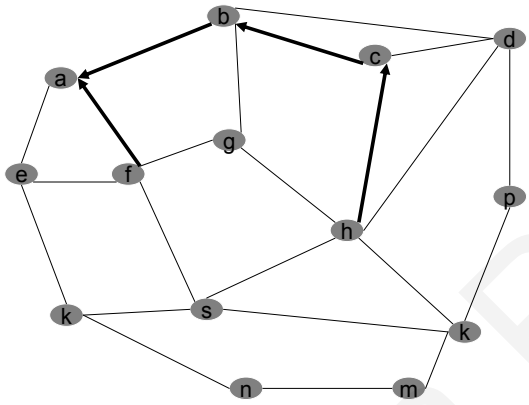

Figure 5.10: Hybrid network G' when only the available physical links of wavelength λ_i are added.

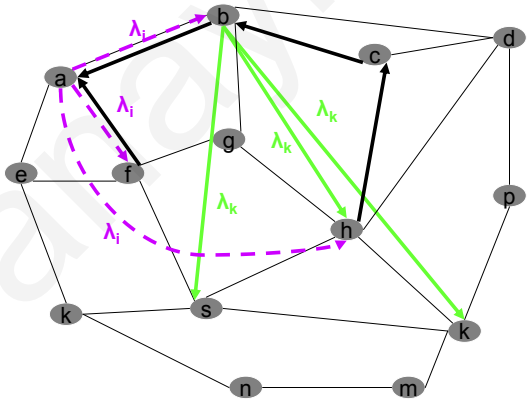

Figure 5.11: Hybrid graph HG.

While in Fig. 5.11 link weights are not shown, logical links are assigned weights that represent their free capacity while physical links are assigned weights that represent their physical distance in order to avoid the creation of long paths that are more susceptible to physical layer impairments. Furthermore, a physical first priority (PP) scheme is followed by multiplying the weights of the logical links by a number that is much greater compared to the maximum link distance. In general, several different schemes can be created by modifying the link weights, such as a logical first priority (LP) scheme or setting the link weights in such a way that physical and logical links are of the same magnitude. However, in this work only the former approach is investigated while a more in-depth investigation of the problem is planned as future work.

The basic steps of the creation of the *HG* for a particular wavelength are described in Algorithm 18 while the description of the HST heuristic used for the routing procedure on *HGs* is also given below.

Algorithm 18 Hybrid Graph Creation

Input: A graph $G = (V, E)$ representing the network, an arriving session with rate k, a list L consisting of the already established light-trees with information about their free capacity, their overlapping degree to the arriving call, the number of calls already serving, their logical links, and their assigned wavelength.
Output: A hybrid graph *HG* for wavelength λ_i.

1: begin
2: $HG \leftarrow G$
3: Remove from *HG* all physical arcs used by λ_i.
4: Remove from L all light-trees with free capacity less than k.
5: Sort light-trees in L according to one of the MCF/MXCF/MXOF/LUF/MUF schemes.
6: Identify number n of light-trees in L.
7: $h \leftarrow n$
8: **while** $h > 1$ **do**
9: **if** light-tree L_h creates at least one logical link with the same end-points as one of the logical links of any light-tree L_i, where $i = 1, ..h-1$ **then**
10: Remove light-tree L_h from list L.
11: $h \leftarrow h - 1$
12: $n = n - 1$
13: **end if**
14: **end while**
15: $h \leftarrow 1$
16: **while** $h \leq n$ **do**
17: **if** none of the logical links in light-tree L_h are already present in *HG* **then**
18: Add logical links of L_h in *HG* with their link weights representing their free capacity.
19: **end if**
20: $h \leftarrow h + 1$
21: **end while**
22: **return** *HG*

5.4.2 Hybrid Steiner Tree Heuristic

The Hybrid Steiner Tree (HST) heuristic is developed for routing on the *HGs*, as known multicast routing heuristics cannot be applied directly on a *HG*. The reason for this is that logical links belonging to the same logical light-tree are grouped together and cannot be separated. Due to transparency and the lack of wavelength conversion in the networks examined, light-trees originating from the same transmitter have to reach every destination node of the light-tree first established for that specific transmitter. Thus, the Steiner Tree (ST) heuristic

described in Algorithm 1 is modified to the Hybrid Steiner Tree (HST) heuristic to account for the inseparability of logical links belonging to the same light-tree. Specifically, the HST heuristic differs from the ST heuristic in that it finds the hybrid multicast tree HT by adding in the currently constructed tree HT the path that leads to the destination node that is closest to HT, but each time a new path is added it has to identify if any of the newly added links corresponds to a logical link. For the links that do correspond to a logical link it identifies their logical light-trees and adds the corresponding logical links to the HT. In each iteration the multicast set is updated by removing from the set all destination nodes added to the HT. The heuristic terminates when every destination node of the multicast call is added to the HT.

Algorithm 19 describes in more detail the basic steps of the HST heuristic. Note that the complexity of the HST heuristic equals in the worst-case scenario the complexity of the ST heuristic, that is $O(kn^2)$, with k representing the multicast group size and n representing the number of nodes in the network under examination.

Algorithm 19 Hybrid Steiner Tree (HST) Heuristic

Input: A hybrid graph $G = (V, E)$ consisting of both physical and logical links, a source node $s \in V$, a destination set $D = [d_1, d_2, ...d_n] \subseteq V$, weights assigned to each edge $e \in E$, and a list L consisting of the light-trees $\in G$.
Output: A hybrid tree HT spanning the set $s \cup D$.

1: begin
2: $T \leftarrow s$, $S \leftarrow s$
3: $k \leftarrow 0$
4: **while** $k < n$ **do**
5: Calculate all shortest paths from node(s) $\in S$ to destination nodes $\in D$
6: Choose the shortest path SP amongst them and add it to tree HT
7: **if** None of the links $\in SP$ correspond to a logical link **then**
8: Identify node $d_j \in D$ last added to tree T
9: Remove destination node d_j from D
10: Add destination node d_j to S
11: $k \leftarrow k + 1$
12: **else**
13: Identify the logical light-trees $\in L$ that correspond to the logical links $\in SP$ and add them to HT
14: Identify node(s) $d_j \in D$ last added to tree HT
15: Remove destination node(s) d_j from D
16: Add destination node(s) d_j to S
17: Identify the number a of nodes $d_j \in D$ last added to tree HT
18: $k \leftarrow k + a$
19: **end if**
20: **end while**
21: **return** HT

Returning to the *HG* of Fig. 5.11 and assuming that the physical links are assigned a weight equal to 1 units while logical links are assigned weights according to their free capacity, the HST heuristic will route the multicast call $R = [e, a, b, p]$ on the *HT* of Fig. 5.12. In this example, the *HT* consists of two logical light-trees and two new physical paths (dotted lines). Note that one of the two logical trees used to build the hybrid tree is not used to serve any of the destination nodes directly but it is used as a bridge to connect one of the destination nodes to the hybrid tree, something that cannot be accomplished by any other algorithm where the physical and the logical networks are treated separately. Regarding the newly created light-trees/lightpaths, their feasibility must first be examined before being established into the network, in terms of availability of transmitters and receivers and in terms of signal quality. In the next section, both the impairment-unaware hybrid provisioning and the impairment-aware hybrid provisioning techniques used for routing/grooming subrate multicast connections are presented.

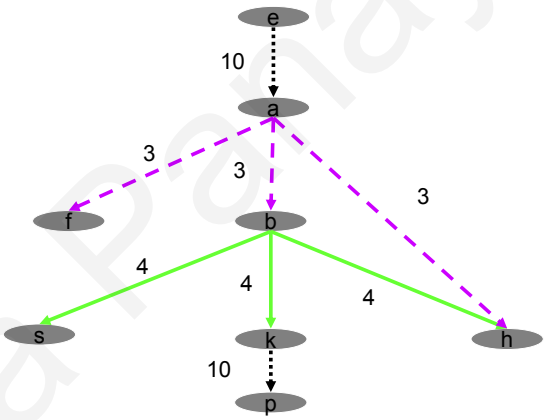

Figure 5.12: Hybrid multicast tree. The dotted lines represent new paths found at the physical layer between the corresponding nodes.

5.4.3 Multicast Connection Provisioning on Hybrid Graphs

For each multicast call, the *Hybrid Multicast Routing and Wavelength Assignment* (HMC-RWA) algorithm builds k hybrid graphs *HG*s, one for each of the k wavelengths in the network. Each *HG* is created according to one of the MCF/MXCF/MXOF/LUF/MUF schemes previously described. The wavelength assignment sub-problem is then solved based on the *first-fit* algorithm while for the hybrid multicast routing sub-problem the HST heuristic is used. Multicast requests are accepted into the network if a hybrid tree exists onto one of the

*k HG*s.

Similarly, for the *Impairment-Aware Hybrid Multicast Routing and Wavelength Assignment* (IA-HMC-RWA) algorithm, for each multicast call the IA-HMC-RWA algorithm builds *k* hybrid graphs *HG*s, one for each of the *k* wavelengths in the network, while *HG*s are created according to one of the MCF/MXCF/MXOF/LUF/MUF schemes previously described. Again, the wavelength assignment problem is solved based on the *first-fit* algorithm while for the hybrid multicast routing problem the HST heuristic is used. Multicast requests are accepted into the network if:

- A hybrid tree exists onto one of the *k HG*s,
- The Q-factor on each destination node of the newly created light-trees is above the predetermined Q-threshold,
- There are available TXs/RXs to support every newly created light-tree.

If the physical impairments are not met, a new wavelength assignment is implemented, in the sense that the a hybrid tree is attempted on the next *HG* according to the list of wavelengths, and the heuristic is repeated until no new wavelength assignment is possible.

5.5 Performance Results

In order to obtain performance results for the proposed grooming techniques, simulations were performed for a metro network consisting of 50 nodes and 98 links (196 arcs), with an average node degree of 3.92 and an average distance between the nodes of 60 Km. Note that the detailed network topology of the metro network used for the simulations is described in Chapter 3. A dynamic traffic model where multicast sessions arrive at each node according to a Poisson process and the holding time is exponentially distributed with a unit mean is used. Thirty-two wavelengths per link were utilized with each wavelength utilizing 10 units of capacity. The rate of each call was randomly generated from the set of integer numbers between 1 and 10. Results were evaluated for both the HMC-RWA and the IA-HMC-RWA algorithms, while results were also evaluated for the PFSR and LFHR techniques [87] for comparison purposes. For the case were the PLIs are considered, a Q-threshold of 8.5 dBQ is assumed corresponding to a BER of 10^{-12} while the node architecture and engineering design shown in Fig. 3.5 and extended in Section 5.2 is assumed.

5.5.1 LFHR and PFSR Schemes

For the PFSR and LFHR schemes the blocking probability versus the multicast group size, for a network load of 100 Erlangs, was evaluated as shown in Figs. 5.13 and 5.14. Specifically, Fig. 5.13 shows the blocking probability versus the multicast group size for the LFHR/PFSR heuristics assuming that only one logical hop is allowed during the logical provisioning phase. Results of the ST heuristic algorithm are also shown for comparison purposes. Results show that the PFSR heuristic outperforms the LFHR heuristic. Furthermore, the LFHR approach for large group sizes (i.e., for multicast destinations close to half of the network nodes), performs worst than the ST heuristic. This is due to the fact that the LFHR heuristic tends to unnecessarily reserve resources in the logical layer, increasing the blocking probability for large group sizes. For example, LFHR may reserve in the logical layer a light-tree that serves only a small portion of the destination nodes in the multicast set, while it would be more efficient to accommodate these nodes in the physical layer. Thus, PFRS that first seeks for a physical route, performs better than LFHR.

To examine whether by increasing the number of allowable logical hops the performance of the PFSR heuristic can be further improved, the blocking probability versus the multicast group size is also evaluated, for a network load of 100 Erlangs, for different number of allowable logical hops. The constraint on the number of logical hops is denoted in Fig. 5.14 with parameter h. Thus, $h = 0$ implies the ST heuristic without considering grooming techniques, $h = 1$ implies the PFSR approach with one logical hop, and $h = 2$ implies the PFSR approach with two logical hops allowed during the logical provisioning phase. Results show that the PFSR technique with $h = 2$ slightly improves the session blocking probability for smaller group sizes, (i.e., for group sizes up to 19), while for large group sizes PFSR with $h = 1$ outperforms the other two cases.

For the rest of the analysis, the number of allowable logical hops for PFSR and LFHR heuristics is set to $h = 1$, since no particular improvement was found if the number of logical hops was increased to 2. A more realistic scenario is also examined where requests of mixed group sizes arrive into the network and the group sizes are randomly generated between the set of integer numbers $[2, 40]$. Hence, multicast as well as unicast requests randomly arrive into the network, with 40 being the maximum multicast group size of a multicast call (all network nodes (minus the source) are destination nodes). The blocking probability versus the load of the network in Erlangs is evaluated as shown in Figs. 5.15 and 5.16. More precisely, Fig. 5.15 shows the blocking probability versus the network load in Erlangs

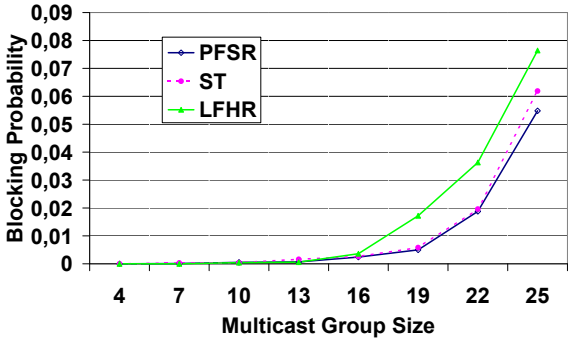

Figure 5.13: Blocking probability versus multicast group size for the PFSR/LFHR schemes. Results of the conventional ST heuristic are also illustrated for comparison purposes.

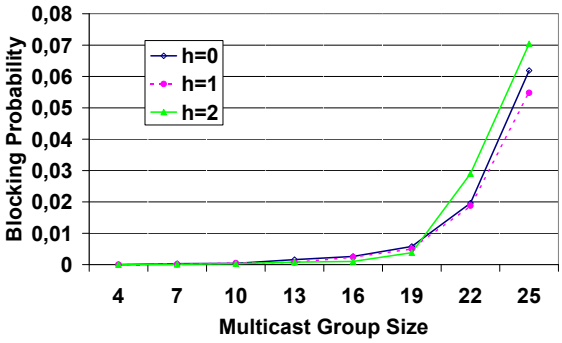

Figure 5.14: Blocking probability versus multicast group size for the PFSR scheme, for different multihop cases ($h = 1$ and $h = 2$). Results of the conventional ST heuristic ($h = 0$) are also illustrated for comparison purposes.

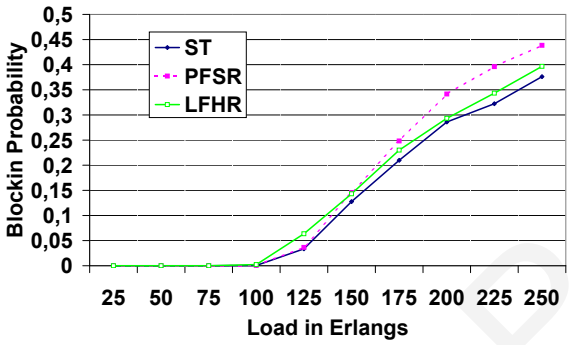

Figure 5.15: Blocking probability versus load in Erlangs for the PFSR/LFHR schemes. Results of the conventional ST heuristic are also illustrated for comparison purposes.

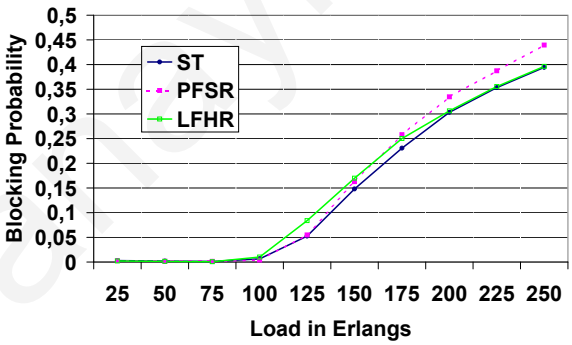

Figure 5.16: Blocking probability versus load in Erlangs for the PFSR/LFHR schemes when the PLIs are also considered. Results of the conventional ST heuristic are also illustrated for comparison purposes.

for the PFSR, LFHR, and ST heuristics without considering any physical layer constraints, while Fig. 5.16 shows the blocking probability versus the network load in Erlangs for the same set of heuristics when the PLIs are also taken into account. As is demonstrated by the results, PLIs affect the blocking probability only for the ST heuristic, in which case the blocking probability is increased. When grooming techniques are considered (PFSR and LFHR) the blocking probability seems to be unaffected when the PLIs are also considered. This is due to the fact that for both the PFSR and LFHR techniques, light-trees that are not entirely feasible in the physical layer can be accommodated in the logical layer thus avoiding the blocking of the request. However, under a different network architecture/engineering, in which the impact of the physical layer effects could be more severe, it is possible that the

PLIs could affect the performance of both the PFSR and LFHR schemes.

Furthermore, according to Figs. 5.15 and 5.16 LFHR is now performing better than PFSR. This is reasonable, as in the latter scenario mixed group sizes result in logical light-trees that can vary in size and thus logical light-trees can now more efficiently accommodate a new call. Unlike before, now it is more likely that a logical light-tree is chosen in which most of the destination nodes are similar to the destination nodes of the arriving call rather than choosing a logical light-tree that serves just a few similar destination nodes in the expense of reserving the capacity of the entire light-tree. In other words, in the latter network scenario, the percentage of the number of similar source/destination nodes to a particular logical tree over the number of all source/destination nodes of the logical light-tree increases, in favor of the LFHR approach. However, results also show that under this network scenario, the performance of both PFSR and LFHR heuristics is worst when compared to the ST heuristic that does not perform grooming. PFSR and LFHR perform better only for low traffic loads (up to 100 Erlangs).

5.5.2 Hybrid Routing/Grooming Schemes

In this section the performance of the newly proposed MCF, MXCF, MXOF, LUF, and MUF hybrid schemes for building the hybrid graphs is examined. For the creation of the hybrid graphs (*HGs*) a physical priority (PP) approach is assumed, during which weights on the *HGs* are set in such a way that the probability of physical resources being chosen first by the hybrid routing algorithm is increased compared to the probability of logical links.

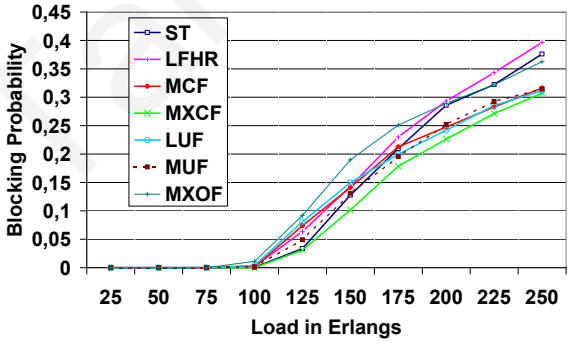

Figure 5.17: Blocking probability versus network load in Erlangs for various hybrid graph creation schemes. Results of the conventional ST heuristic and the LFHR scheme are also illustrated for comparison purposes.

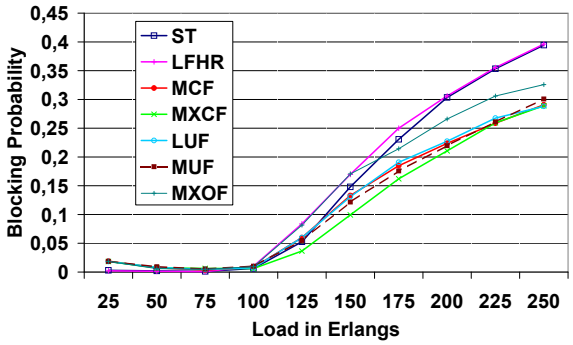

Figure 5.18: Blocking probability versus network load in Erlangs for various hybrid graph creation schemes when the PLIs are also considered. Results of the conventional ST heuristic and the LFHR scheme are also illustrated for comparison purposes.

For these simulations, requests of mixed group sizes again arrive into the network and the group sizes are randomly generated between the set of integer numbers $[2, 40]$. Hence, multicast as well as unicast requests randomly arrive into the network, with 40 being the maximum multicast group size of a multicast call. Fig. 5.17 shows the performance results for the MCF, MXCF, MXOF, LUF, and MUF heuristics when compared to the LFHR and ST heuristics. Note that for LFHR scheme the number of allowable logical hops is again set to $h = 1$. According to Fig. 5.17, hybrid routing schemes improve the blocking probability as the load increases, with the MXCF scheme performing the best. Results for the MCF/LUF/MUF schemes are very similar, while only the MXOF scheme performs worst than the conventional ST heuristic. The results of the MXOF scheme show that creating the hybrid graphs according to the overlapping degree of the arriving call to the existing logical light-trees, does not yield an efficient hybrid topology of available resources. Results of Fig. 5.17 show that it is more efficient to give priority to logical links with more free capacity to be placed first on the HG.

However, as shown in Fig. 5.18, when the PLIs are also considered all hybrid schemes perform better than the LFHR and ST heuristics, with MXCF performing the best and MXOF resulting in the worst blocking probability amongst the hybrid schemes. Compared to the case where the PLIs were not considered (Fig. 5.17), the blocking probability is only slightly increased for traffic loads up to 100 Erlangs where multicast calls arrive and leave the network very quickly. The blocking probability in this case is increased due to the fact that grooming in such light loads results in the utilization of resources for a period of time that is much greater compared to the actual time that these resources would have been utilized if no grooming was performed. This usage of resources, forces the HST heuristic to create light-trees that have more depth and thus leading to an increased blocking probability due to a low Q-factor. For larger loads however, the blocking probability of the hybrid schemes is improved even though the PLIs are considered. This is caused due to the fact that now shorter trees are computed in the physical layer, resulting in logical light-trees that utilize fewer resources. Thus, the HST works better since each time it adds to the hybrid tree a destination node that is connected through a logical link, it adds to the tree less unnecessary logical links.

Finally, it is interesting to note that the second best hybrid scheme, is the most-used light-tree first (MUF) that gives priority to the logical light-tree that is being used by the maximum number of calls to be placed first on the HG. This is mainly due to the fact that in this way the blocking probability due to TXs/RXs is decreased compared to the least-used light-tree

first (LUF) technique that has less probability of finding available TXs/RXs to accommodate the newly arrived multicast connection.

5.6 Conclusions

In this work, the multicast traffic grooming problem under physical layer constraints is examined. A hybrid routing algorithm is proposed, namely the hybrid Steiner tree (HST) heuristic that routes multicast calls on hybrid graphs, consisting of both the available physical and logical links. Several schemes are proposed for building the *HGs* with each scheme prioritizing the logical resources according to a different characteristic. MXCF/MCF schemes prioritize logical links according to their free capacity, MUF/LUF according to the number of calls logical links are already serving, and MXOF according to the overlapping degree among the existing light-trees and the arriving call. The HST heuristic in conjunction with the aforementioned schemes, is compared to the conventional ST heuristic as well to the existing LFHR/PFSR heuristics that route the new multicast calls on logical and physical layers separately and sequentially.

Results show that HST heuristic outperforms existing approaches that were developed here for comparison purposes, for every hybrid scheme proposed, especially when the PLIs are also considered. Amongst the hybrid schemes proposed, MXCF outperforms the rest, with MXOF exhibiting the highest blocking probability.

In general, increased blocking probability for the PFSR/LFHR heuristics is caused due to the fact that available physical and logical resources are treated separately and sequentially and although physical or logical resources may exist to serve the new call, these resources are either not able to be utilized by PFSR/LFHR or cannot be combined in the most efficient way. In hybrid schemes, however, the HST heuristic can have global information of the availability and connectivity of physical and logical links, thus making it possible for the resources to be utilized in a more efficiently way. Amongst the hybrid schemes proposed, the MXCF scheme performs the best which leads to the conclusion that it is more efficient to give priority to channels that have more free capacity to be utilized first. The maximum overlapping light-tree first (MXOF) scheme performs the worst amongst the hybrid schemes proposed, leading to the conclusion that giving priority to light-trees that can serve the largest number of nodes in a multicast set is less important when compared to schemes that prioritize logical links according to their free capacity or the number calls that are already serving.

Chapter 6

Groupcasting in Transparent WDM Mesh Optical Networks

Apart from the unicast and multicast applications discussed in the previous chapters, next-generation high-bandwidth networks are also expected to support even more bandwidth-intensive applications, such as groupcast applications that now require each node in a set of nodes to send information to all other nodes in the set by for example now creating a set of light-trees, one for each node as the source. As wavelength-routed networks are deployed in great numbers to meet scalability and bandwidth requirements in next-generation networks, groupcast connections will be even more commonly utilized in optical networks to serve multipoint-to-multipoint bandwidth intensive sessions. Many applications that require groupcast can include grid-computing, multi-party teleconferencing, distributed interactive simulations, virtual private network (VPN) services and Ethernet LAN (E-LAN) services. For such applications to be viable, it is important that the groupcast traffic is routed efficiently through the optical network and it is also protected against any possible failures in the network. Failure protection against link failures is very significant in these applications, since groupcast sessions carry traffic to multiple destinations and the impact of a link failure is even more severe compared to the impact of the same failure on a unicast or multicast session. Furthermore, as grooming techniques increase the bandwidth utilization of an optical network by grooming low-speed traffic streams onto high-speed wavelengths channels, it is essential to apply grooming for groupcast sessions that require a huge amount of bandwidth to be established.

To support groupcast sessions in a transparent optical network, optical splitters must again be used inside the nodes to split the incoming signal to multiple output ports. There-

fore, optical amplifiers are again required in the network to compensate for the power losses the signals experience prior to reaching their destinations. As the optical amplifiers also introduce noise to the signal (ASE noise), this impairment, along with the rest of the PLIs described in Chapter 2, must again be taken into consideration when solving the groupcast routing/grooming and wavelength assignment, as well as the groupcast protection problems.

This chapter extends the unicast and multicast work presented in the previous chapters by now dealing with the provisioning and protection of groupcast sessions in transparent WDM mesh optical networks when the physical layer impairments are now also taken into account. There is very little body of work on optical groupcasting and no work as of now on groupcasting with physical layer impairment considerations. This is the first time that groupcast connection provisioning and protection techniques are examined under physical layer constraints.

In this chapter, the groupcast routing and wavelength assignment problem is first investigated and a number of groupcast routing schemes are presented. Their performance is compared and examined when the PLIs are taken into account. Also, the protection of groupcast sessions is examined and several groupcast protection algorithms are presented and their performance is also evaluated when the PLIs are considered. Finally, the groupcast traffic grooming problem is investigated and a number of grooming techniques for groupcast sessions is presented and analyzed for both the case where the PLIs are not considered and the case where the PLIs are taken into account.

6.1 State-of-the-art

In general, groupcasting refers to a multipoint-to-multipoint connectivity. In an optical network, a light-forest [140] is constructed to serve a groupcast session, which is a set of light-trees or lightpaths. In a dynamic system where groupcast requests arrive and leave the network dynamically, the objective of the groupcast routing and wavelength assignment (GC-RWA) problem is to reduce as much as possible the session blocking probability given a fixed number of wavelengths. The assumption in most works in the literature is that a groupcast request is accepted in the network if a route and a wavelength assignment is possible for the entire groupcast session, otherwise it is blocked. Finding the routes for a groupcast session is equivalent to finding a set of multicast trees or point-to-point paths in such a way that every node in the groupcast session can send information to every other node in the session and receive information from every other node in the session simultaneously. In

the absence of all-optical wavelength conversion, and for transparent optical networks, the wavelength continuity constraint must be met during the wavelength assignment procedure. Since the RWA problem was proven to be NP-complete for lightpath [59] and light-tree [84] establishment, the GC-RWA problem is also an NP-complete problem [142].

Relevant work in the area of groupcasting in WDM optical networks can be found in [28, 30, 31, 140, 142, 143, 159–161, 214]. Optical groupcasting is a new research subject and there is only a handful of papers that have only barely started to look at this problem, even though there is substantial work that has been performed on groupcasting in other research areas. Authors in [28, 30, 31, 159–161, 214] investigated the optical groupcast traffic grooming problem in WDM networks, where subrate connections are present in the network. The groupcast traffic grooming problem was formulated mathematically via integer linear programming and heuristic algorithms were also proposed and investigated under various network topologies and traffic conditions. Furthermore, authors in [140, 142, 143] investigated the static GC-RWA problem for several design approaches aiming at minimizing the total network cost, as well as minimizing the number of wavelengths utilized in the network. Several lightpath and light-tree-based groupcast RWA algorithms were proposed and their performance was evaluated for OXC-based and MC-OXC-based networks, respectively.

6.2 Groupcast Routing and Wavelength Assignment Algorithm

The dynamic GC-RWA problem can be decomposed into the two sub-problems, namely the groupcast routing problem (GC-R) and the groupcast wavelength assignment (GC-WA) problem, and then each problem can be solved separately (and sequentially). The GC-R problem corresponds to finding a set of point-to-point paths or multicast trees in such a way that every node in the groupcast set can reach every other node in the set and can be reached from every other node in the set as well. After the GC-R problem is solved, different techniques can then be utilized to find the wavelength assignment for each path or tree in the groupcast session, including any of the wavelength assignment techniques detailed in previous chapters of this thesis (first-fit, most-used, least-used, etc).

In the point-to-point (path) based approach [142], if the groupcast set consists of k nodes, then $k\times(k-1)$ point-to-point paths need to be computed. For example, the groupcast set $GC = (d_1, d_2, d_3)$ consisting of three nodes, is decomposed into six point-to-point (unicast) sets

$U_1 = (d_1, d_2)$, $U_2 = (d_1, d_3)$, $U_3 = (d_2, d_1)$, $U_4 = (d_2, d_3)$, $U_5 = (d_3, d_1)$, and $U_6 = (d_3, d_2)$. Then, for each point-to-point set a specified routing algorithm (e.g., Dijkstra's shortest path algorithm) can be used for the calculation of each path. In the light-tree approach [142], if the groupcast set consists of k nodes, then k multicast trees need to be routed. For example for a groupcast set $GC = (d_1, d_2, d_3)$ consisting of three nodes, three multicast trees need to be computed; one for each multicast set $MC_1 = (d_1, d_2, d_3)$, $MC_2 = (d_2, d_1, d_3)$, and $MC_3 = (d_3, d_1, d_2)$, where the first node in the multicast set corresponds to the source node of the tree and the rest of the nodes in the multicast set correspond to the destination nodes. In transparent optical networks, optical splitters can be used in network nodes to split the incoming signal to multiple outputs, thus enabling the establishment of connections with multiple destinations [156]. As pointed out in previous chapters, finding the multicast tree corresponds to solving the Steiner Minimum Tree (SMT) problem that is NP-complete, and therefore several heuristics have been developed to approximately solve the problem [174, 185].

In this dissertation, four groupcast routing algorithms were developed and compared:

- *Light-Tree Heuristic:* The groupcast request is decomposed into a set of multicast requests and each multicast tree is computed using the Steiner Tree (ST) heuristic described in Algorithm 1. For each multicast group it finds the minimum cost tree, while the link costs represent the physical distance between the nodes.

- *Lightpath Heuristic:* The groupcast request is decomposed into a set of point-to-point requests and each path is computed using Dijkstra's shortest path algorithm.

- *Linear Light-Tree Heuristic:* The groupcast request is decomposed into a set of multicast requests and the computation of each linear light-tree is based on the Optimized collapsed ring (OCR) approach proposed in [88]. This approach starts with a tree T that has only one member, the source node. Then it adds to the tree the path from the source to its shortest destination node. The last node added to the tree is then considered as the "source node", and the heuristic is repeated until every destination node is added to the tree.

- *k-Light-Trees Heuristic:* In this approach, the groupcast request is decomposed into a set of multicast requests and each multicast tree is computed using the k-Steiner Tree (k-ST) heuristic. For each multicast group it performs k trials and in each trial it calculates the minimum cost tree T_i using the ST heuristic described in Algorithm 1.

After the end of each trial, a link belonging to T_i is randomly chosen and its cost on the corresponding graph network is set to ∞. By doing so, a different multicast tree is computed in each iteration of the algorithm. The minimum cost tree amongst the k possible trees is then chosen. Table 6.1 describes the basic steps of the k-ST heuristic on a network graph G. Note that the link costs in G represent the actual physical distances between the network nodes.

Table 6.1: Basic Steps of the k-ST Heuristic.

Step 1	$i = 1$
Step 2	$G'' = G$.
Step 3	In G'', calculate the minimum cost tree T_i using the Steiner Tree (ST) heuristic described in Algorithm 1.
Step 4	Pick randomly a link $e_i \in G''$ in such a way that $e_i \in T_i$.
Step 5	$G' = G$.
Step 6	Set cost of link $e_i \in G'$ to ∞ and create graph G''.
Step 7	$i = i + 1$.
Step 8	If $i > k$ then go to Step 9, otherwise go to Step 3.
Step 9	Choose the minimum cost tree, amongst the possible T_i trees, where $i = 1, 2, 3..., k$.

The wavelength assignment sub-problem, in the absence of wavelength conversion, was also proven to be NP-complete (as it can be transformed into the vertex coloring problem in graph theory, a known NP-complete problem) and several heuristics have been developed on assigning wavelengths to the routes (or trees). In any wavelength assignment heuristic the objective is to minimize the number of wavelengths utilized by the light-forest with the constraint that a single wavelength must be assigned to each point-to-point path or multicast tree from the source node to every destination node. For example, Fig. 6.1 shows a light-tree based connection for the groupcast set $GC = (1, 2, 3, 4)$. In this example, four light-trees are required for supporting the session, utilizing three wavelengths. In general, to arbitrate wavelength contentions among lightpaths/light-trees, a random node sequence is generated for the multicast group. The first node generates its route and requests optical channel(s) for its route, publishing them to the second node. The second node generates its route and considers optical channels available, excluding the requested channels from the first node; the third node considers channels excluding the first and second route requests. For the $k \times (k-1)^{th}$ path or k^{th} tree, wavelength channels available are only those left after $[k \times (k-1) - 1]$ paths or $k - 1$ trees respectively, reserve their channels [142].

During the GC-RWA algorithm used in this work, each wavelength in the network is viewed as a different layer of the network. Each layer is a replica of the initial network

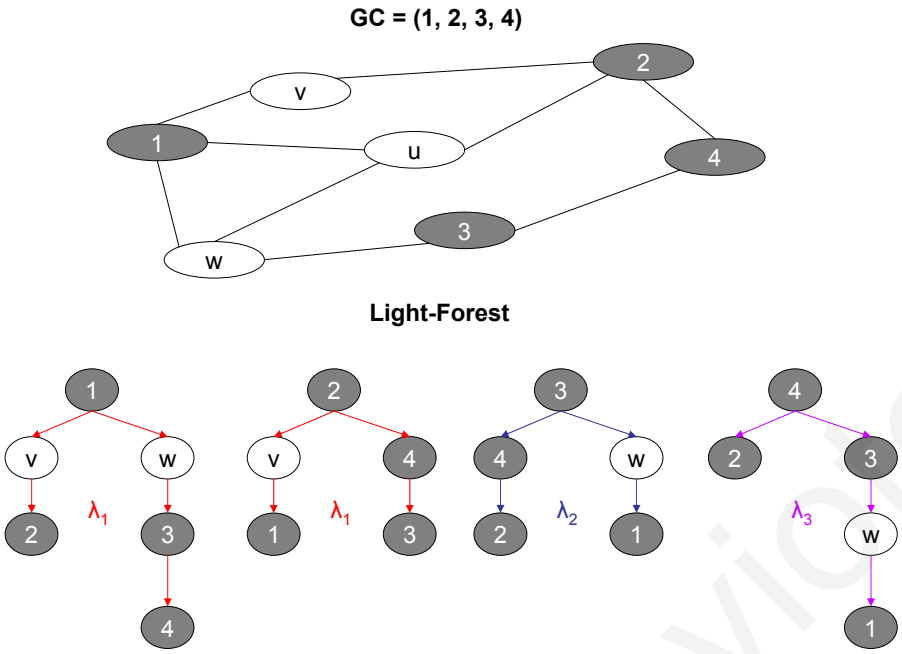

Figure 6.1: Light-tree based optical groupcast.

topology which is updated each time a request is established/released to/from the network. Therefore, each time a request is established in the network on a certain wavelength, the corresponding network layer (wavelength graph) is updated in such a way that the links occupied by the request are removed from the corresponding network layer (wavelength graph) and are added back upon the departure of the request. When a groupcast request arrives into the network, the routing decisions are made based on the current network state. A groupcast request is blocked if no wavelength assignment is possible for at least one of the paths/multicast trees in the set. Fig. 6.2 describes the GC-RWA algorithm developed in this work for the light-tree based approach. Each time a groupcast request arrives into the network, it is decomposed into the appropriate number of multicast sets. Then for each multicast set, the MC-RWA heuristic algorithm described in Chapter 3 is used. If the algorithm fails to find a light-tree for at least one multicast set the entire groupcast group is blocked (the assumption in this work is that all multicast sets must be established for the groupcast session to be provisioned in the network). Otherwise, it is accepted into the network. Each time a light-tree is computed, the corresponding wavelength links are marked as unavailable in order to be excluded from the calculation of the next light-trees. If the groupcast request is blocked, wavelength links for every previously computed light-tree in the same session are

released for the calculation of the next groupcast request.

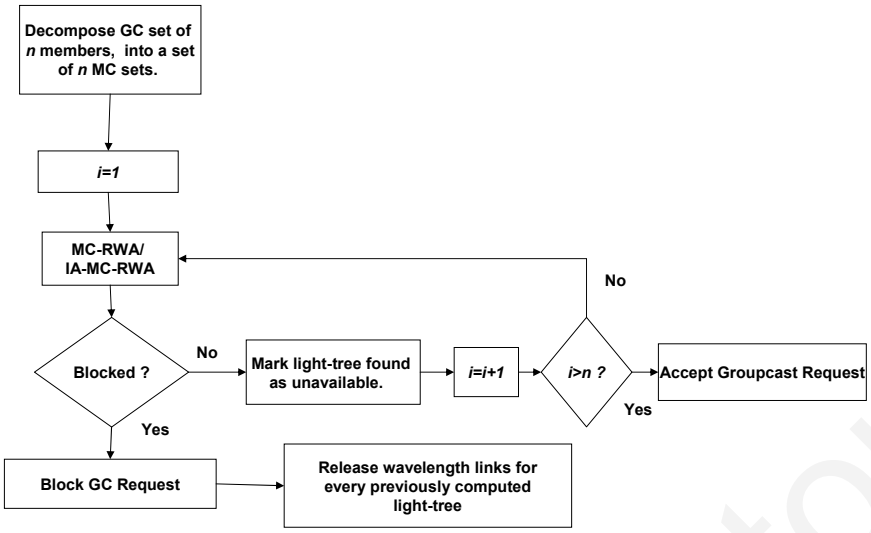

Figure 6.2: Flowchart of GC-RWA/IA-GC-RWA algorithm.

In a transparent optical network, in order for a connection to be feasible, apart from the wavelength continuity constraint, the PLIs must also be accounted for. As pointed out, the impairments introduced by the physical layer may degrade the signal to the point where the signal is not detectable at the receiver end. Therefore, before establishing a connection into the network it is important to ensure that the Q-factor at every destination node is above the predetermined Q-threshold. The physical layer system model described in Chapter 2 is also used here, for the calculation of the Q-factor. In the impairment-aware GC-RWA (IA-GC-RWA) algorithm, a groupcast request is accepted into the network if:

- There is an available wavelength for every light-tree/lightpath in the groupcast set.
- The Q-factor for every destination node in each light-tree/lightpath in the groupcast set is above the predetermined Q-threshold.
- There are available TXs/RXs for every light-tree/lightpath in the groupcast set.

The IA-GC-RWA algorithm developed in this work for the light-tree approach, is also shown in Fig. 6.2. For every multicast set, the IA-MC-RWA algorithm described in Chapter 3 is now used to account for the PLIs.

6.3 Groupcast Protection Techniques

In this section, the protection of groupcast sessions upon any single link failure is investigated. Since groupcast sessions may carry traffic to multiple destinations, the impact of a

link failure is even more severe compared to the impact of a link failure on a unicast or multicast session. Therefore, it is essential for the groupcast traffic to be routed and also to be protected in an efficient manner.

The key objective of a groupcast protection algorithm is to ensure that every affected destination can receive the information from the source via the backup path(s) after the failure. In general, protection of groupcast sessions corresponds to finding a primary light-forest and its backup light-forest. As pointed out, a light-forest can be a set of lightpaths or light-trees. Since the light-tree approach performs better that the lightpath approach as shown in [142], in this work only the former case is considered. For a groupcast request with n members denoted as $GC = (d_1, d_2, ..., d_n)$, its primary light-forest is denoted as $LF = (T_1, T_2, .., T_n)$, where T_i is the primary light-tree for multicast set $MC_i = (d_i, D_i)$. Note that d_i is the source node of the i^{th} multicast tree while the rest $n-1$ member nodes are included in the destinations set D_i. If a primary light-forest LF can be provisioned and can also be protected, the groupcast request is accepted in the network; otherwise it is blocked. Thus, the objective of this work is to devise protection algorithms that minimize the resources that are used to protect the primary light-forest LF and therefore the blocking probability in the network.

One solution to the groupcast protection problem is to directly use known multicast protection schemes for the creation of the backup light-forest $LF' = (T'_1, T'_2, ..., T'_n)$, where T'_i corresponds to the backup paths of the primary light-tree T_i. This approach, however, does not take into consideration the unique characteristics of the groupcast sessions. Thus, in this work two novel groupcast protection algorithms are developed that now also take into account the unique characteristics of the light-forests corresponding to the groupcasting sessions. The basic idea behind both heuristics relies on the fact that between any multicast sets in a forest the member nodes are the same. Thus, if any two multicast trees can be packed in the same wavelength and protected via the same backup path, then the number of resources used for protecting the entire light-forest is reduced, and so is the blocking probability.

Specifically, in the first scheme called cycle-for-two (CFT), a single cycle that passes through every member in the groupcast request is computed to support two multicast trees in the case of a single link failure, while in the tree-for-two (TFT) scheme a new tree is computed to support two multicast trees. The constraint in both schemes is that primary trees must be arc-disjoint from each other and from their backup paths as well.

As pointed out, known multicast protection algorithms could also be directly adapted for solving the groupcast protection problem. This work investigates the groupcast protection problem by examining the performance of such protection algorithms and by also comparing

them to the CFT and TFT heuristic algorithms. Specifically, CFT and TFT heuristics are developed and compared to several multicast protection heuristics, namely the Arc-Disjoint Trees (ADT), the Modified Conventional Segment-Based Protection (MCSP), the Modified Segment-Based Protection With Sister Node First (MSSNF) and the Level Protection (LP) heuristic algorithms described in Chapter 4. The proposed heuristics are shown to obtain better performance results compared to the rest of the protection techniques, with the CFT heuristic performing the best amongst all of them.

6.3.1 Cycle-for-two Heuristic Algorithm

In the CFT approach, a single cycle that passes through every member in the groupcast request is computed to support two multicast trees in the case of a single link failure. The constraint is that primary trees must be arc-disjoint from each other and from their backup paths as well. Table 6.2 describes the basic steps of the CFT heuristic, between any two multicast sets of a groupcast session. Note that the protection cycle is calculated first, since in this way the savings in terms of network resources are increased.

Table 6.2: Basic Steps of the CFT Heuristic for Multicast Sets MC_1 and MC_2 of Groupcast Session GC.

Step 1	In G, compute a linear light-tree c_1 starting from any node in groupcast set GC and visiting exactly once every other node in set GC, using the OCR (optimized collapsed ring) heuristic [141].
Step 2	Remove the arcs of c_1 from G and create graph G'.
Step 3	In G', compute shortest path c_2 connecting the end-points of c_1 and starting from the node last added to c_1, using a shortest path (Dijkstra's) algorithm.
Step 4	Merge c_1 and c_2 onto protection cycle C.
Step 5	Remove the arcs of c_2 from G' and create graph G''.
Step 6	In G'', find two primary arc-disjoint trees T_1 and T_2 for MC_1 and MC_2, using the Steiner Tree (ST) heuristic.

Figs. 6.3 and 6.4 are used as an illustrative example of the CFT heuristic for groupcast set $GC = (a, b, c, d)$. Specifically, Fig. 6.3 shows only the two of the four multicast trees that must be created for the groupcast session, originating from source nodes a and b. In the same figure, multicast trees $MC_1 = (a, b, c, d)$ and $MC_2 = (b, a, c, d)$, computed by the Steiner Tree (ST) heuristic, are protected via the same cycle in such a way that both trees and the protection cycle are arc-disjoint from each other. Fig. 6.4 shows how the two light-trees are reconfigured upon the failure of link (a, b) by using the appropriate arcs of the protection cycle. Similar to any other protection scheme in which backup paths are shared

between several primary paths, the protection arcs that will be utilized are not known prior to the failure and therefore an automatic protection switching (APS) protocol is required for setting-up the new light-trees after a link failure has occurred.

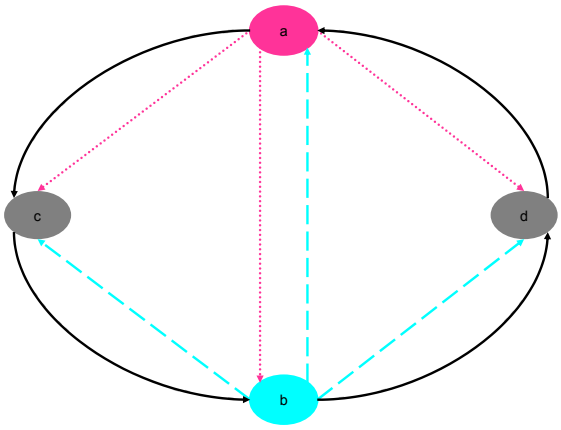

Figure 6.3: A cycle passing through every node in $GC = (a, b, c, d)$ supports multicast sets $MC_1 = (a, b, c, d)$ and $MC_2 = (b, a, c, d)$.

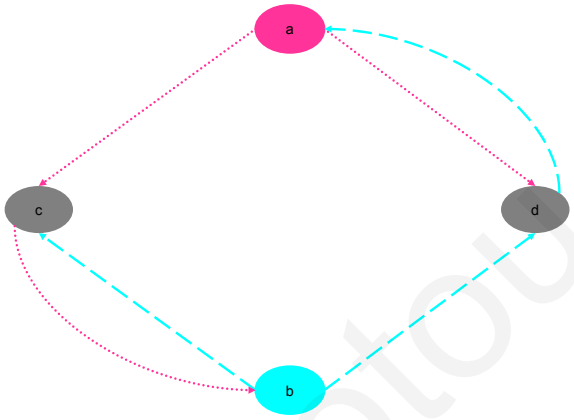

Figure 6.4: Protection paths upon failure of link (a, b) in Fig. 6.3.

6.3.2 Tree-for-two Heuristic Algorithm

In the TFT approach, a new tree is computed to support two multicast trees. The constraint again is that primary trees must be arc-disjoint from each other and from their backup paths as well. Table 6.3 describes the basic steps of the TFT heuristic between any two multicast sets of a groupcast session. Again, the protection tree is calculated first, since in this way the savings in terms of network resources are increased.

Figs. 6.5 and 6.6 are used to demonstrate the basic idea of the TFT heuristic for groupcast set $GC = (a, b, c)$. Specifically, Fig. 6.5 shows only two of the three multicast trees that must be created for the groupcast session, originating from source nodes a and b. In the same figure, multicast trees $MC_1 = (a, b, c)$ and $MC_2 = (b, a, c)$ computed by the Steiner Tree (ST) heuristic are protected via the same multicast tree that originates from a random node v in the network, in such a way that all three trees are arc-disjoint from each other. Fig. 6.6 shows how the two light-trees are reconfigured upon the failure of link (a, b) by using the appropriate arcs of the protection tree. As discussed before, an APS protocol is again required for rerouting the traffic after the failure has occurred. Note that the protection tree starts from a random node in the network and spans every other node in the groupcast set. Since performing an exhaustive search for finding the best source for the protection

Table 6.3: Basic Steps of the TFT Heuristic for Multicast Sets MC_1 and MC_2, of Groupcast Session GC.

Step 1	In G, calculate two arc-disjoint shortest paths p_1 and p_2 using a shortest path algorithm (e.g., Dijkstra's), connecting the source nodes of multicast sets MC_1 and MC_2 in both directions.
Step 2	Remove from set GC the source nodes of multicast sets MC_1 and MC_2. Remove the arcs of paths p_1 and p_2 from G to create graph G'.
Step 3	In G', calculate a multicast tree T_1 using the Steiner Tree (ST) heuristic, starting from any node in paths p_1 and p_2 and spanning every node in set GC.
Step 4	Identify the source node s' of tree T_1. If $s' \in p_j$, where $j = 1$ or 2, then calculate shortest path p_3 from any node in path p_i, where $i = 1$ or 2 and $i \neq j$, to node s'.
Step 5	Remove arcs of tree T_1 and path p_3 from graph G' to create graph G''.
Step 6	Merge tree T_1 and paths p_k, where $k = 1, 2, 3$ onto protection tree T.
Step 7	In G'', find two primary arc-disjoint trees T_2 and T_3 for MC_1 and MC_2 respectively, using the ST heuristic.

tree increases the computational complexity of the algorithm, the TFT heuristic, as shown in Table 6.3, tries to identify a convenient source node v in a few algorithmic steps. Note that in cases where there is an odd number of members of a groupcast set, then in both the CFT and TFT heuristics the remaining multicast set is protected separately utilizing the ADT protection technique.

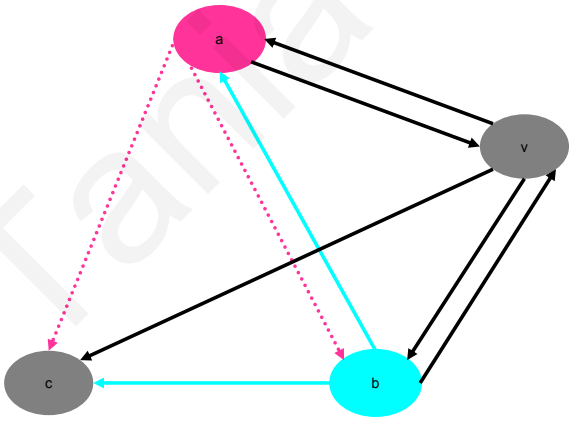

Figure 6.5: Tree-for-two example: A tree passing through every node in $GC = (a, b, c)$, supports two light-trees for $MC_1 = (a, b, c)$ and $MC_2 = (b, a, c)$, where the first node in the sets corresponds to the source node.

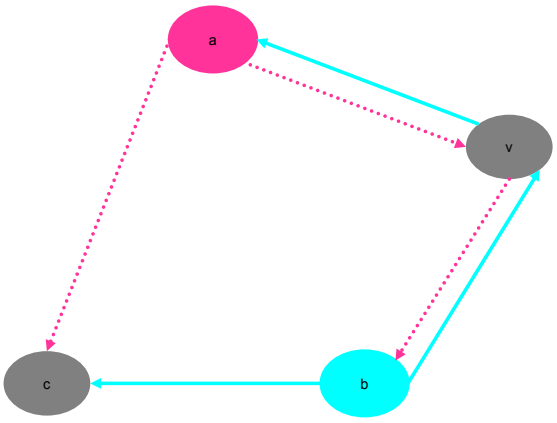

Figure 6.6: Tree-for-two example: Reconfiguration of light-trees upon failure of link (a, b) in Fig. 6.5.

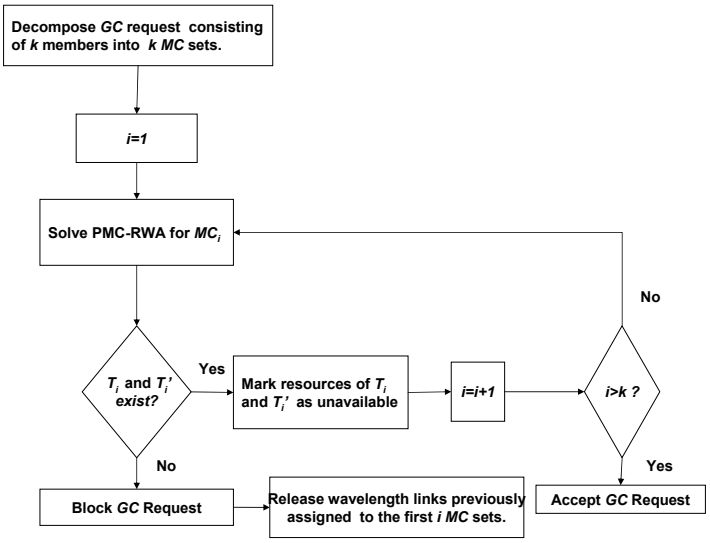

Figure 6.7: Flowchart of the protected groupcast routing and wavelength assignment (PGC-RWA) algorithm when one of the ADT, MCSP, MSSNF, or LP [107, 129, 173, 198] heuristic algorithms is used for the computation of the backup paths.

6.3.3 Provisioning of Protected Groupcast Sessions

During the *impairment-unaware provisioning*, for each groupcast request GC consisting of k members, the protected groupcast routing and wavelength assignment (PGC-RWA) algorithm breaks the GC set into k MC sets. For each MC_i set the protected multicast routing and wavelength assignment (PMC-RWA) algorithm described in Chapter 4 is applied. Specifically, for the protected multicast routing sub-problem, a primary light-tree T_i and its backup paths T_i' on the same wavelength are found. The Steiner Tree (ST) heuristic [174] is used for the computation of the primary light-trees while for the computation of their backup paths one of the ADT, MCSP, MSSNF, or LP heuristics described in Chapter 4 is used. For the wavelength assignment procedure the *first-fit* algorithm is utilized. Groupcast requests are blocked if there is no available wavelength for the entire primary tree and its backup paths. The flowchart of Fig. 6.7 describes in detail the PGC-RWA algorithm used in this work.

The aforementioned approach is slightly modified when the CFT or TFT heuristics are used, in order to pack multicast sets into pairs in a random fashion before the computation of the primary tree and its backup paths. Specifically, for each groupcast request GC consisting of k members, the PGC-RWA algorithm breaks the GC set into k MC sets. Subsequently, it randomly packs the multicast sets into $k/2$ MC pairs and for each MC pair the PMC-RWA algorithm is solved. For the protected multicast routing sub-problem, one of the proposed CFT or TFT heuristic algorithms is used while for the wavelength assignment procedure the first-fit algorithm is again utilized. Groupcast requests are blocked if there is no available

wavelength for all primary trees and their backup paths. The flowchart of Fig. 6.8 describes in detail the modified PGC-RWA algorithm used for the CFT and TFT heuristic algorithms. Note that, the flowchart of Fig. 6.8 assumes that the number of the k members included in the GC set is even. However, if k is an odd number, then one more algorithmic step is required for the remaining MC set. For the remaining MC set, the PMC-RWA algorithm is used, during which the ADT heuristic is utilized during the protected multicast routing sub-problem and the first-fit algorithm is utilized during the wavelength assignment procedure.

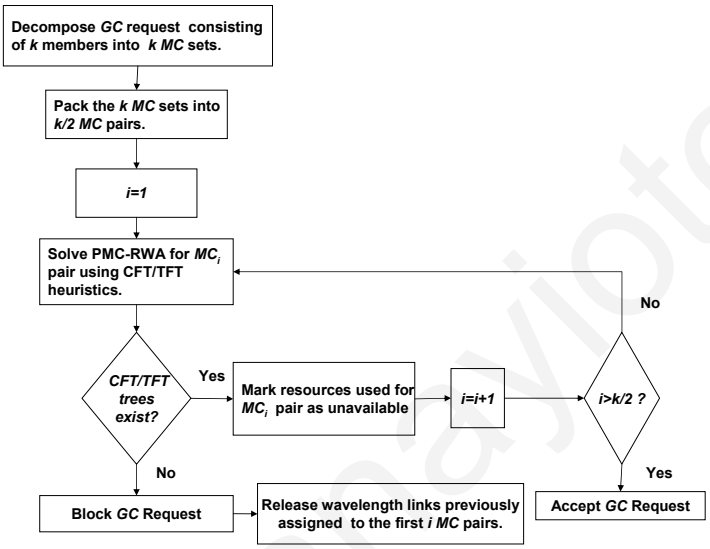

Figure 6.8: Flowchart of the protected groupcast routing and wavelength assignment (PGC-RWA) algorithm when one of the CFT or TFT heuristic algorithms is used.

During the *impairment-aware provisioning*, once a groupcast request arrives into the system, the impairment-aware protected groupcast routing and wavelength assignment (IA-PGC-RWA) algorithm decomposes the groupcast request into its proper multicast sets. Then, for each multicast set (or pair of multicast sets in the case of CFT and TFT heuristics), the IA-PGC-RWA algorithm first solves the multicast routing and wavelength assignment problem by finding a primary light-tree and its backup paths on the same wavelength using the first-fit wavelength assignment algorithm. The light-tree(s) and its backup paths are considered feasible if:

- A wavelength assignment is possible for the entire light-tree(s) and its backup path(s).
- The Q-factor of each path on the primary light-tree(s) and on its backup path(s) is above the predetermined Q-threshold.
- There are available TXs/RXs for that connection.

If the constraint for the physical impairments of the primary light-tree(s) and/or the backup paths is not met, a new wavelength assignment is implemented and the heuristic is repeated until no new wavelength assignment is possible for that connection. Groupcast requests are blocked if at least one multicast set or a pair of multicast sets is not feasible. Otherwise, the groupcast request is accepted into the network. Table 6.4 describes the basic steps of the IA-PGC-RWA algorithm for a groupcast request that is decomposed into k multicast sets. Note that if the CFT or TFT heuristics are assumed, Table 6.4 refers to a groupcast request with the multicast sets packed into k multicast pairs.

Table 6.4: IA-PGC-RWA Algorithm

Step 1	Set $i = 0$.
Step 2	$i = i + 1$.
Step 3	If $i > k$ go to Step 7, or otherwise go to Step 4.
Step 4	Find the primary light-tree(s) and its backup paths using the IA-MC-RWA algorithm on multicast set i.
Step 5	If set i is feasible declare as unavailable the wavelength links that are used for set i and go to Step 2. Otherwise go to Step 6.
Step 6	Groupcast request is blocked.
Step 7	Groupcast request is accepted.

6.4 Groupcast Traffic Grooming Algorithms

Applying traffic grooming techniques for sessions of groupcast connectivity is essential due to the high demand on bandwidth for the groupcast connections. In general, grooming techniques increase the bandwidth utilization of an optical network, by grooming low-speed traffic streams onto high-speed wavelength channels. As in current network deployments typical connection requests do not require the full wavelength capacity, it is only reasonable to multiplex several independent lower-speed traffic streams onto a single lightpath or light-tree in order to efficiently utilize the capacity of each wavelength channel. In this section, the groupcast traffic grooming problem in transparent optical networks is investigated when the PLIs are also considered.

In networks where groupcast requests arrive and leave the network dynamically, the objective of the Groupcast Routing/Grooming and Wavelength Assignment problem is to minimize the session blocking probability given a fixed number of wavelengths. The physical first sequential routing (PFSR) scheme was developed amongst others in Chapter 5. While in Chapter 5 the PFSR heuristic was developed for multicast sessions, in this section PFSR is

extended for groupcast connections as well. The objective of this work is to evaluate the performance of multicast grooming heuristics when these are extended for groupcast sessions, and when the PLIs are also taken into account. As previously shown, the PFSR heuristic performs logical and physical provisioning phases sequentially, by considering the physical and logical layers of the network separately. In the PFSR scheme, a physical route is searched for first, and if this fails the logical provisioning phase takes place.

The *logical provisioning* phase corresponds to finding a groupcast grooming algorithm that efficiently utilizes the capacity of each wavelength channel. The maximum overlapping light-tree (MOL) heuristic presented in Chapter 5 is thus developed for the logical provisioning phase. Simulation results in Chapter 5 show that when the MOL heuristic is used in conjunction with the PFSR approach, the blocking probability is decreased compared to the case were no grooming techniques were considered. Since a groupcast request consisting of k members can be decomposed into k multicast sets, the MOL heuristic can be used as described in Table 5.1 for each multicast set separately. The difference here, is that once the logical paths are decided for a multicast set, the state of the logical topology is updated in order for the capacity used for the current multicast set to be seen as unavailable upon the computation of the logical paths of the next multicast set.

The *physical provisioning* phase corresponds to finding a set of multicast trees or point-to-point paths in such a way that every node in the groupcast session can send information to every other node in the session and receive information from every other node in the session simultaneously. Three groupcast routing algorithms are described in this chapter. Results in [142] show that the Light-Tree heuristic, performs the best compared to the lightpath and the Linear Light-Tree heuristics. Thus, only the Light-Tree heuristic will be considered here. During the physical provisioning phase, the Impairment-Aware GC-RWA problem must be solved, which is equivalent to solving the IA-MC-RWA algorithm for each multicast set that the groupcast request is decomposed to. For each multicast set, the IA-MC-RWA algorithm first solves the routing sub-problem according to the sub-Steiner tree (SST) heuristic described in Algorithm 17 and then assigns a wavelength for that route. The multicast request is feasible, if:

- A route and a wavelength assignment is possible for the entire multicast tree.
- The Q-factor for each path on the multicast tree is above the predetermined Q-threshold.
- There are available TXs/RXs for that connection.

If the physical impairment constraints are not met, a new wavelength assignment is implemented and the heuristic is repeated until no new wavelength assignment is possible. Then, the multicast set is updated by removing from its multicast set the destination nodes that are already included in the sub-light-tree. The MOL heuristic is performed according to the updated multicast set and the groupcast request is blocked if MOL fails to accommodate every destination node in the updated multicast set. Note that if during the physical provisioning phase the entire multicast set is served, the logical provisioning phase is not performed.

The IA-MC-RWA algorithm is performed sequentially for every multicast set and if it fails to accommodated at least one multicast set, then the entire groupcast request is blocked. The heuristic terminates after k successive iterations or earlier if it fails for a multicast set. Table 6.5 describes the basic steps of the PFSR heuristic in conjunction with the MOL heuristic for a groupcast request consisting of k members.

Table 6.5: PFSR with MOL Heuristic

Step 1	$i = 0$.
Step 2	$i = i + 1$.
Step 3	If $i > k$, go to Step 9.
Step 4	Perform the IA-MC-RWA for the multicast set MC_i and return updated multicast set MC_i'.
Step 5	If $MC_i' = \emptyset$, then go to Step 2, otherwise go to Step 6.
Step 6	Perform MOL for MC_i', allowing for h logical hops, and return the updated multicast set MC_i''.
Step 7	If $MC_i'' = \emptyset$, then go to Step 2, otherwise go to Step 8.
Step 8	Groupcast request is blocked.
Step 9	Groupcast requests is accepted.

6.5 Performance Results

In order to obtain performance results for the groupcast techniques, simulations were performed utilizing the node architecture and engineering design of Fig. 3.5 (described in Chapter 3), with passive optical splitters and fixed TXs/RXs. In order to evaluate the average performance of the algorithms, groupcast connections are simulated on a metropolitan optical network consisting of 50 nodes and 98 links (196 arcs), with an average nodal degree of 3.92 and an average distance between the nodes of 60 Km. Note that more details on the network used in the simulations can be found in Chapter 3.

A dynamic traffic model was also utilized, where groupcast sessions arrive in the network

according to a Poisson process and the holding time is exponentially distributed with a unit mean. The network load is set at 50 Erlangs. In each simulation 2,000 requests were generated for each groupcast group size. For each heuristic the blocking probability versus the multicast group size is evaluated, and the Q-threshold is set at 8.5 dBQ which corresponds to a BER of 10^{-12}.

6.5.1 Routing Schemes

To evaluate the blocking probability versus the groupcast group size for the groupcast routing schemes presented, sixty-four wavelengths were utilized for each fiber. Results were evaluated for the case where the PLIs are not taken into account and the case where the PLIs are considered as a validation step of the Q-factor at the destinations nodes after the routing and wavelength assignment of the connection requests. Specifically, during the IA-GC-RWA algorithm that accounts for the PLIs (Fig 6.2), if a light-forest exists, the Q-factor for each path on the light-forest is evaluated and the groupcast request is blocked if there is at least one path on that light-forest with a Q-value that falls below the predetermined Q-threshold, or if there are no available TXs/RXs for the entire light-forest, and there is no alternate wavelength assignment possible. Otherwise, a new wavelength assignment is implemented and the heuristic is repeated until no new wavelength assignment is possible.

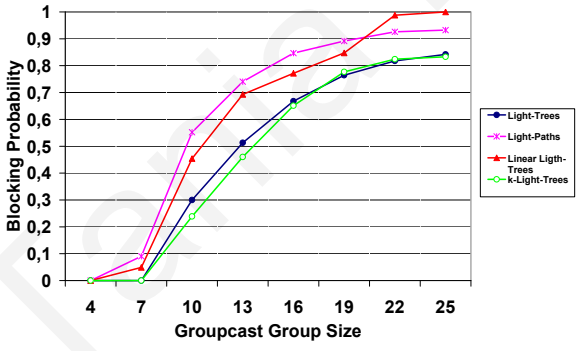

Figure 6.9: Blocking probability versus groupcast group size for different groupcast routing schemes.

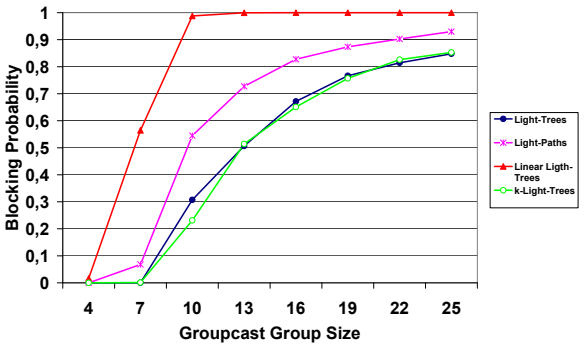

Figure 6.10: Blocking probability versus groupcast group size when the PLIs are also considered for different groupcast routing schemes.

Figs. 6.9 and 6.10 show the blocking probability versus the groupcast group size when the PLIs are not taken into account and when the PLIs are considered respectively for different groupcast routing schemes. In both cases the Light-Tree approach performs the best compared to the Lightpath and Linear Light-Tree approaches. Furthermore, the k-Light-

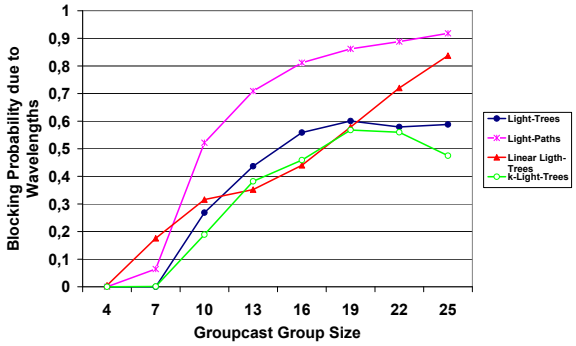

Figure 6.11: Blocking probability due to the unavailability of wavelengths versus groupcast group size when PLIs are considered, for different groupcast routing schemes.

Trees heuristic with $k = 10$, improves the blocking probability compared to the conventional Light-Trees approach where $k = 1$. When the PLIs are also taken into account, the blocking probability is clearly increased only in the Linear Light-Trees heuristic, while for the other heuristics the blocking probability seems to be unaffected by the PLIs (Fig. 6.10). Fig. 6.11 shows the blocking probability due to the unavailability of wavelengths versus the groupcast group size where it is clearly illustrated that while a portion of the blocking probability is caused due to the low Q-factor at the destination nodes the blocking probability is mainly limited by the unavailability of wavelengths. The reason that the overall blocking probability when the PLIs are considered seems to be unaffected in the presented heuristics, is that when some sessions are blocked due to low Q-factor the blocking probability due to the lack of wavelengths is decreased since more wavelengths remain available. In the Linear Light-Trees approach, blocking probability is significantly affected by the PLIs, since in this case the computed light-trees tend to be very long in distance and thus increasing the impact of the PLIs on the signal when it reaches the destination nodes.

In all cases, however, the blocking probability is significantly high. The k-Light-Trees heuristic that aims at minimizing the cost of the tree, achieves the lowest blocking probability, an indicator that it is important for groupcast routing algorithms to have as an objective the minimization of the cost of the entire light-forest.

6.5.2 Protection Schemes

To evaluate the performance of the different protection algorithms described in this chapter, 128 wavelengths per fiber were utilized to evaluate the blocking probability versus the groupcast group size. If a primary light-forest can be provisioned and can also be protected,

the groupcast request is accepted in the network; otherwise it is blocked.

Fig. 6.12 shows the blocking probability versus the groupcast group size for a number of groupcast protection heuristics. Results show that the CFT heuristic outperforms the rest of the protection schemes, while the TFT heuristic only slightly outperforms the rest of heuristic algorithms that were initially developed for multicast connections and only for small group sizes. Furthermore, it is clear that both the CFT and TFT heuristics achieve much lower redundant capacity for protecting the groupcast sessions compared to the rest of the heuristics, as in these two cases each cycle found protects all the links for a pair of multicast trees. However, when cross-sharing is also considered in the routing algorithms, the LP heuristic outperforms the rest, while the CFT and TFT heuristics perform the worst. This is shown in Fig. 6.13. The reason for this is that the CFT and TFT heuristics perform sharing of their backup resources between two primary trees by default, while the other heuristics reserve a number of resources for one primary tree. Thus, when cross-sharing is also considered, the blocking probability of the ADT, MCSP, MSSNF, and LP heuristics significantly improves, while the blocking probability of the CFT and TFT heuristics is only slightly improved. While, the ADT, MCSP, MSSNF, and LP heuristics share their backup resources in a more flexible way, CFT and TFT have a strict way of sharing their resources thus resulting in backup paths that utilize a larger number of links compared to the rest of the schemes. Note that, according to Fig. 6.13, the LP heuristic is now outperforming the rest of the protection schemes, that were initially proposed for multicast protection purposes.

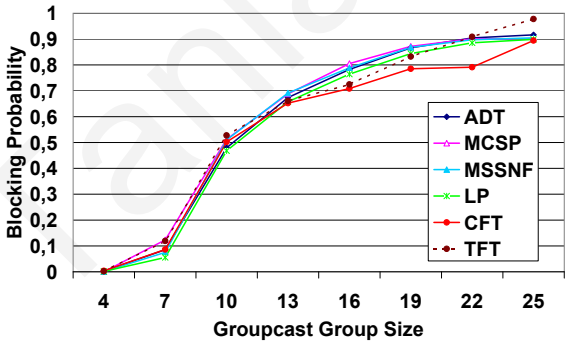

Figure 6.12: Blocking probability for protected groupcast sessions versus groupcast group size.

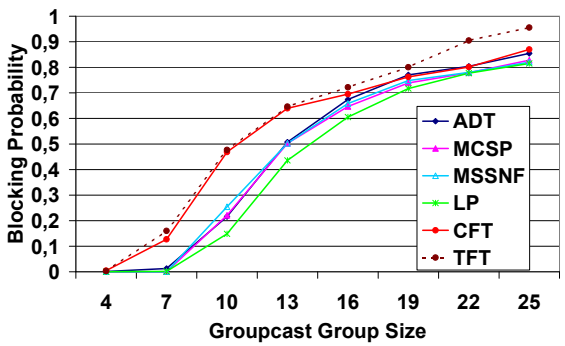

Figure 6.13: Blocking probability for protected groupcast sessions versus groupcast group size when cross-sharing is considered.

Figs. 6.14 and 6.15 show the blocking probability for the protected groupcast sessions versus the groupcast group size when the PLIs are also considered. In Fig. 6.14 cross-sharing is not considered, while in Fig. 6.15 cross-sharing is performed amongst the backup

resources of the different primary light-trees. In both cases, the LP heuristic performs the best. The blocking probability of the ADT, MCSP, MSSNF, and LP heuristics is now only slightly affected by the PLIs while the blocking probability of the CFT and TFT heuristics is dramatically increased due to long backup paths that both heuristics tend to create. Again, Fig. 6.15 shows that when cross-sharing is used, the blocking probability is improved.

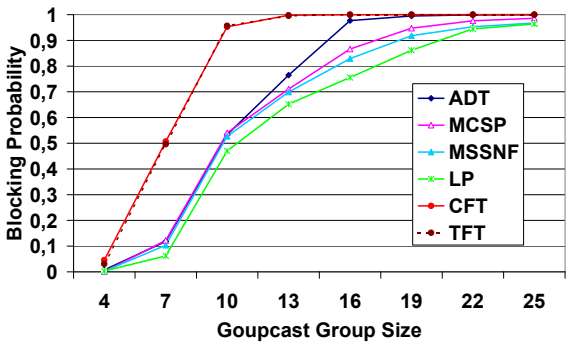

Figure 6.14: Blocking probability for protected groupcast sessions versus groupcast group size with PLIs.

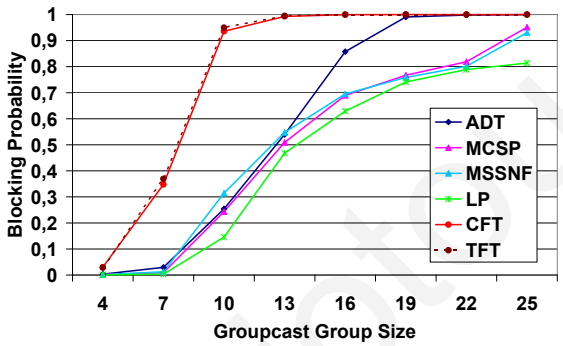

Figure 6.15: Blocking probability for protected groupcast sessions versus groupcast group size with cross-sharing and with PLIs.

According to the above results, the CFT and TFT heuristics perform better that the other heuristics that were developed here for comparison purposes only when the PLIs are not taken into consideration and cross-sharing is not performed in a global way in the network. Note that cross-sharing of the resources in general increases the computation complexity of the algorithms since a more complex mechanism is required in the network in order to keep information of the reserved resources and their capabilities of being shared. Furthermore, even though the CFT and TFT heuristics by default share their backup resources, simpler APS techniques need to be developed to arbitrate the common resources in case of a link failure compared to the case where cross-sharing is performed in a more random way. In general, however, when dealing with transparent optical network, the PLIs must be considered, and when they do, the CFT and TFT heuristics are inefficient due to the long backup paths that tend to create. In this case, the LP heuristic proposed in this work in Chapter 4 performs the best for both multicast and groupcast connections.

6.5.3 Grooming Schemes

For the performance evaluation of the groupcast grooming techniques, sixty-four wavelengths per link were utilized, with each wavelength utilizing 10 units of capacity, and the rate of each call was randomly generated between the set of integer numbers 1 and 10.

Fig. 6.16 shows the blocking probability versus the groupcast group size for the PFSR technique when the MOL heuristic was used during the logical provisioning phase. Different values were set for parameter h that indicates a constraint in the number of hops in the logical layer for the MOL heuristic. During the physical provisioning phase, the Minimum Hop Tree (MHT) heuristic was used. Note that MHT is a modification of the ST heuristic and aims at decreasing the number of hops from the source node to the destination nodes and consequently the number of total links used in the tree. The implementation of the MHT heuristic is the same as the ST heuristic, having as the only difference the input of the algorithm, where the costs on the links of the network are set to the same value (e.g., cost=1), instead for the actual distances of the links. The specific algorithm is chosen here, since previously shown results indicated that under the current traffic load and a fixed number of wavelengths, the blocking probability is mainly limited by the number of links utilized by the tree and not by the minimum (in distance) tree, even when the PLIs are considered. Thus, for comparison purposes, Fig. 6.16 also shows the results of the MHT heuristic without the consideration of any grooming techniques.

Performance results show that the MOL approach with $h = \infty$, that allows for an unlimited number of possible logical hops performs the best compared to the other cases. Note that the simulation results of multicast traffic grooming in Chapter 5 show that allowing for just a single hop is the most efficient approach, while when increasing parameter h the results become slightly worst compared to the case were no grooming techniques were performed. However, when dealing with groupcast connections, which require more bandwidth than multicast connections, grooming techniques become more vital.

Results were also evaluated for the case where the PLIs are considered as shown in Fig. 6.17. Specifically, Fig. 6.17 shows the blocking probability versus the groupcast group size for the MHT heuristic algorithm and for the PFSR heuristic with $h = \infty$. Results shows that the blocking probability is not affected by the PLIs since, under this network scenario, the blocking probability is mainly limited by the unavailability of wavelengths.

6.6 Conclusions

In this chapter, groupcast routing/grooming and protection problems in transparent WDM optical network are investigated. As very little work exists in the literature that investigates groupcast connections, this chapter serves as a first insight to the aforementioned problems, when considering the physical layer constraints when designing the routing/grooming and

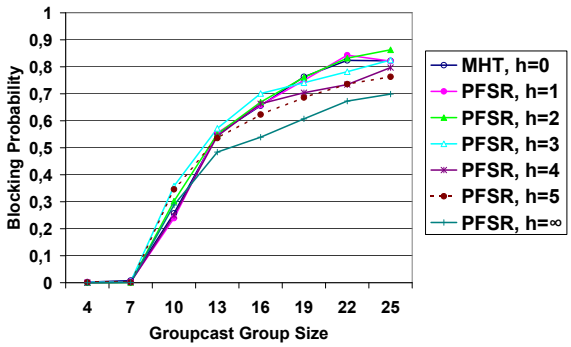

Figure 6.16: Blocking probability for groupcast grooming techniques versus groupcast group size.

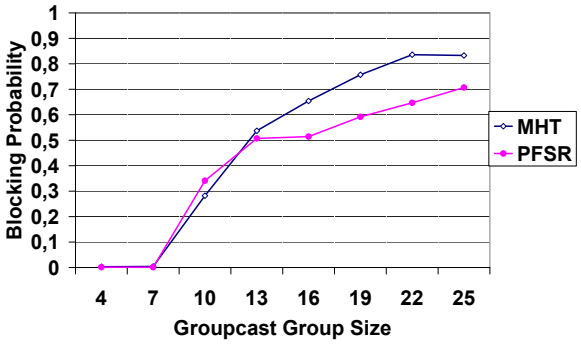

Figure 6.17: Blocking probability for groupcast grooming techniques versus groupcast group size when the PLIs are also considered.

protection heuristics.

The performance of several groupcast routing schemes was evaluated on a metropolitan optical network with and without PLIs. Results showed that for such applications, that require a huge amount of bandwidth, the major limitation is the available number of wavelengths in the network. However, the PLIs must also be considered for the design of the algorithms since the creation of long paths can degrade the signal quality and dramatically affect the system performance. The k-Light-Trees heuristic that aims at minimizing the cost of the tree, achieves the lowest blocking probability, an indicator that it is important that the groupcast routing algorithms must have as a goal the minimization of the cost of the entire light-forest.

Furthermore, the performance of several groupcast protection schemes was evaluated when the PLIs were also considered during the provisioning phase of the connections. Specifically, several protection schemes initially proposed for multicast traffic were extended to support groupcast connections and two new protection schemes, namely the CFT and TFT heuristics, that consider the unique characteristics of groupcast connections, were also developed. Note that even though every algorithm proposed for the routing/grooming or for the protection of multicast sessions, can be extended in order to support groupcast connections, these algorithms do not consider the unique characteristics of groupcast connections. As pointed out, algorithms proposed specifically for groupcast sessions take advantage of the fact that between any multicast set of the same groupcast session, all members are the same and the only difference between them is that a different member node is declared as the source node in each multicast set.

Results showed that while CFT and TFT heuristics generally outperform the other approaches for the protection of groupcast connections, however, when cross-sharing techniques and the PLIs are considered, the LP heuristic that was first proposed in Chapter 4 for multicast connections outperforms all of the other approaches.

Finally, to investigate the groupcast traffic grooming problem a heuristic algorithm initially developed for multicast connections was extended to support groupcast connectivity. Specifically, the PFSR technique in conjunction with the MOL heuristic developed for the logical provisioning phase were compared to conventional groupcast routing algorithms. Several multi-hop cases were also examined, since in the MOL heuristic a constraint can be set for the maximum number of logical hops (h) allowed. Results showed the necessity of performing grooming in such bandwidth intensive applications, since the blocking probability was significantly improved especially for large values of h. When the PLIs were also considered results were not affected, since under the specific network scenarios examined the blocking probability is mainly limited by the number of available resources in the network rather than the physical layer impairments.

Chapter 7

Conclusions and Future Directions

7.1 Conclusions

The main focus of this dissertation was to design the network nodes, engineer the network, and derive algorithms and techniques so as to efficiently provision lambda (and sub-lambda) unicast, multicast, and groupcast connections in all-optical metropolitan area networks and protect them against failures, while at the same time taking into consideration the physical layer impairments in these networks (Quality-of-Transmission (QoT) based provisioning and protection techniques). The main motivation behind this work as discussed extensively in Chapter 1 is the evolution of current and next-generation optical networks from opaque towards translucent and eventually transparent optical networks (TONs) where the signal stays entirely in the optical domain without undergoing any optical-to-electrical-to-optical (OEO) conversions. These transparent networks are extremely desirable as they provide bit-rate, protocol, and modulation format transparency, thus providing better solutions in the implementation of the network architecture by minimizing the extra cost, power, and footprint associated with the additional transceivers/transponders present in an opaque architecture.

This dissertation fills a void in the existing literature on transparent optical networks that includes works that have examined provisioning and protection techniques for transparent optical networks, while at the same time considering the physical layer impairments, but have barely looked at one-to-many connections. Specifically, the few techniques in the literature that investigated provisioning multicast connections with QoT considerations only looked at the power of the optical signal at the destinations. The work presented in this thesis outlines for the first time a complete treatment for this problem, investigating node architectures, network engineering, as well as the design of a large number of heuristic algorithms

that solve in an efficient manner the problem of provisioning and protecting one-to-many connections in TONs.

The thesis achieved all its objectives as highlighted in Chapter 1. New architectures, algorithms, and techniques were developed that provision multicast and groupcast approaches while considering a host of physical layer impairments and at the same time keeping the blocking probability for these connections as low as possible. Furthermore, simple protection techniques were designed that are implemented and precomputed prior to the failure occurrence that enable the network to recover from any single link failure while at the same time keeping the recovery time and redundant protection capacity as low as possible. In addition to the recovery time and redundant capacity, the proposed techniques also ensure that the signal that will reach the receivers after the recovery process is implemented will have an acceptable signal quality, something that was missing from previous research work on multicast protection techniques.

In Chapter 2 an introduction to the existing routing and wavelength assignment problem (RWA) for point-to-point connections is given which is followed by the description of the most important PLIs that affect the signal quality in an all-optical network. The existing work in the literature on the impairment-aware routing and wavelength assignment (IA-RWA) problem is analyzed and then the physical layer system model that is used throughout the thesis is presented in detail via a mathematical formulation, a computational example, and through simulations. Specifically, the IA-RWA algorithm developed in this work is compared to the conventional RWA algorithm (which does not consider any physical layer impairments) and performance results showed a significant increase in the blocking probability for the IA-RWA techniques. This is an indicator of the importance of considering the PLIs during the provisioning phase of the requests and is part of the motivation for the work presented in this thesis. As initial work on the RWA problem did not consider the interaction between the physical and logical layers, the performance results of these approaches were not very practical in a real network implementation. The development of the IA provisioning techniques on the other hand can provide the network designers and operators with practical tools for the provisioning of real network applications in transparent optical networks.

Chapter 3 deals with the provisioning of multicast connections in transparent optical networks, and more precisely with the multicast routing and wavelength assignment problem (MC-RWA). This dissertation improves on the existing MC-RWA algorithms by also including physical layer constraints utilizing the Q-factor. Several multicast routing heuristic algorithms are proposed that also consider the various physical layer impairments present

in the network when the optical signal is transported to the destinations via multicast light-trees. Also, several multicast-capable node designs and network engineering solutions are proposed including node architectures utilizing passive or active splitters, and node architectures with different TXs/RXs designs. The impact of node design/network engineering on the algorithms for multicast routing is shown, as well as the impact of the physical layer constraints, via the Q-factor, on the multicast routing algorithms. The effect of PDG/PDL is also considered in Chapter 3. Results demonstrated an increased complexity that renders the conventional probabilistic handling of PDG/PDL not practical and the need for more refined and computationally efficient interaction between physical/control layers.

Chapter 3 showed that different node architectures, different engineering of the physical layer, and different multicast routing heuristic algorithms result in different network performance results, a strong indicator that a more refined interaction between physical and logical layer is needed for multicast connection provisioning. This dissertation presented efficient node architecture designs, network engineering solutions, and efficient QoT multicast provisioning heuristic algorithms and techniques that can provide realistic solutions for transparent optical networks. The proposed QoT multicast provisioning heuristic algorithms and techniques outperform all routing approaches that do not consider the physical layer effects, or routing algorithms that only account for the power budget in terms of blocking probability, the most important metric when provisioning dynamic connections in the network.

Following the provisioning work in Chapter 3, Chapter 4 examines the problem of protecting multicast connections in transparent optical networks. As pointed out, in wavelength routed networks with high-bandwidth fiber links, it is not only important to route efficiently a connection request but also to maintain the survivability of the connection when link failures occurs. Survivability is even more critical in multicast connection provisioning since multiple destinations may be affected upon a single failure. In this thesis, a novel segment-based protection heuristic algorithm, called level protection (LP) was proposed and compared to other conventional dedicated and segment-based protection techniques. The different protection techniques are compared for the case when the PLIs are considered and the case when the PLIs are not taken into account. Results show that the LP algorithm performs the best, especially when cross- and self-sharing techniques are also utilized, and the PLIs are taken into consideration. This is a clear indicator that when designing protection techniques the effect of PLIs cannot be ignored for solutions that require quality-of transmission (QoT) guarantees. The solution presented is simple, it provides very fast recovery from the failure (on the order of a few hundreds of milliseconds) since the protection paths are precomputed

prior to the failure event, and is capacity efficient as sharing of the redundant capacity takes place amongst several primary paths.

Apart from the segment-based protection solution, Chapter 4 presents additional cycle-based protection heuristic algorithms that are based on the p-cycle concept. The proposed approach now sets a limit on the length of cycles created by utilizing the constraints of the physical layer. This approach was shown to exhibit the best network performance compared to other traditional cycle-based protection techniques. This is the first time cycle-based techniques are investigated taking into consideration the physical-logical layer interactions, making the results extremely important, as they show the infeasibility for some of the most popular cycle-based protection techniques that appear in the literature.

Furthermore, the cycle-based schemes were compared to tree- and segment-based schemes and results showed that cycle-based schemes perform better than tree-based schemes only for large group sizes and are outperformed by segment-based schemes, an indicator that segment-based approaches are the preferred choice for these types of applications. However, it is important to note that segment-based schemes require complex protection protocols for the signaling and switching mechanisms, thus increasing their recovery time compared to the high recovery speeds for the cycle-based protection schemes. It is also noted that when sharing techniques are involved, a more complex switching mechanism is required and the link that has failed needs to be identified, since different link failures lead to different backup paths. Therefore, new protection protocols need to be developed on the signaling and switching mechanisms when sharing techniques are considered.

As most multicast service applications require capacity that is only a fraction of the wavelength capacity (sub-wavelength connections), it is essential that the sub-wavelength traffic demands are allocated to lightpaths or light-trees such that the recourses are shared, to efficiently utilize the capacity of each wavelength channel. In Chapter 5 the multicast traffic grooming problem is investigated, where several multicast traffic grooming techniques are developed and compared. The PLIs are also taken into account for the design of the multicast traffic grooming algorithms and a grooming capable architecture as well as a proper network engineering design are proposed. A hybrid routing algorithm is proposed in this dissertation that routes multicast calls on hybrid graphs consisting of both the available physical and logical links. Several schemes are proposed for building these hybrid graphs, with each scheme prioritizing the logical resources according to a different characteristic. The hybrid routing heuristic in conjunction to the above schemes, is compared to existing grooming heuristics that route the new multicast calls on logical and physical layers separately and sequentially.

Results showed that the proposed techniques outperform the existing approaches for every hybrid scheme proposed, especially when the PLIs are considered.

Groupcast connectivity was finally investigated in Chapter 6, as an extension to the multicast problem, motivated by the recent emergence of several groupcast applications. In Chapter 6, groupcast routing/grooming and protection problems in transparent WDM optical networks are investigated. As only a few works exist in the literature that investigate groupcast connections, the work in this dissertation serves as a first insight to the aforementioned problems, when designing groupcast routing/grooming and protection heuristics with physical layer constraints consideration. The performance of several groupcast routing schemes proposed in the literature was evaluated on metro networks and results showed that for such applications, that require a very large amount of bandwidth, the major limitation is the available network resources (e.g., number of wavelengths in the network). However, the PLIs must also be considered in the design of these algorithms since the creation of long paths can degrade the signal quality and dramatically affect the system performance. In this chapter groupcast routing algorithms were developed that reduce the connection cost while at the same time achieving acceptable signal quality at the receivers. Groupcast grooming techniques were also examined as an extension of the multicast grooming techniques of Chapter 5 and results showed the necessity for performing grooming in such bandwidth intensive applications. In this case the effect of the PLIs was not very pronounced, mainly due to the fact that the network performance was in a great extend limited by the number of available resources in the network.

Several protection schemes, initially proposed for multicast traffic, were extended in this chapter to support survivable groupcast connections, and subsequently new protection schemes that consider the unique characteristics of groupcast connections were developed. The performance of the groupcast protection schemes was evaluated and results showed that the proposed groupcast protection schemes outperform the protection schemes initially proposed for multicast sessions. However, when the PLIs were also considered during the provisioning phase of the working and backup connections and cross-sharing was used, the LP heuristic that was initially proposed for multicast sessions in Chapter 4 outperforms all other approaches.

7.2 Future Directions

During the course of the thesis work several other interesting problems were encountered. However, since not all of the problems can be addressed in the thesis, there are noted in this section as topics of future work.

One possible direction for future work is to include into the physical layer system model described in this thesis the crosstalk effect in a more accurate way, rather than including it as a budget value into the final Q-factor. As pointed out, in this work crosstalk is considered only for the new requests, through a Q-budgeting approach, and not for the connections already established into the network. However, each time a new connection is added into the network, existing connections may be also affected by inter-channel crosstalk in the fiber links or by co-channel crosstalk in the nodes (switches, MUXs, DMUXs). Thus new connection requests should be added into the network only if the existing ones are not affected.

Another direction for future work could be the design of optical networks considering mixed line rates and multiple modulation formats. With the growth of traffic volume and the emergence of various new applications, future telecom networks are expected to be increasingly heterogeneous with respect to applications supported and underlying technologies employed. To address this heterogeneity, it may be cost effective to set up different lightpaths/light-trees at different bit rates, as low-bit-rate services will need less grooming (i.e., less multiplexing with other low-bit-rate services onto high-capacity wavelengths), while a high-bit-rate service can be accommodated directly on a wavelength. Furthermore, the unregenerated reach of a lightpath depends on its line rate. So, the assignment of a line rate to a lightpath/light-tree is a tradeoff between its capacity and transparent reach. Thus, based on their signal-quality constraints, intelligent assignment of line rates to connections can minimize the need for signal regeneration. This constraint on the transparent reach based on threshold signal quality can be relaxed by employing more advanced modulation formats, but with more cost. For example, some long-distance, high-bit-rate paths could use an improved modulation format having higher bandwidth-distance product and being less susceptibility to impairments [124].

One other avenue for future work is the addition of a small number of regeneration points inside the transparent optical network in order to improve the performance of the system (breaking the network into "islands of transparency"). Regeneration may take place inside a multicast-capable node or at the network links. One possible solution to the design of multicast-capable nodes with signal regeneration is shown in Fig. 7.1. According to this

design, a number of TXs and RXs are placed just after the DMUX in order for the pass-through channels to be regenerated. Thus, if N is the number of available channels in the network and M is the fan-out of the node, $(N \times M)$ RXs and $(N \times M)$ TXs will be required.

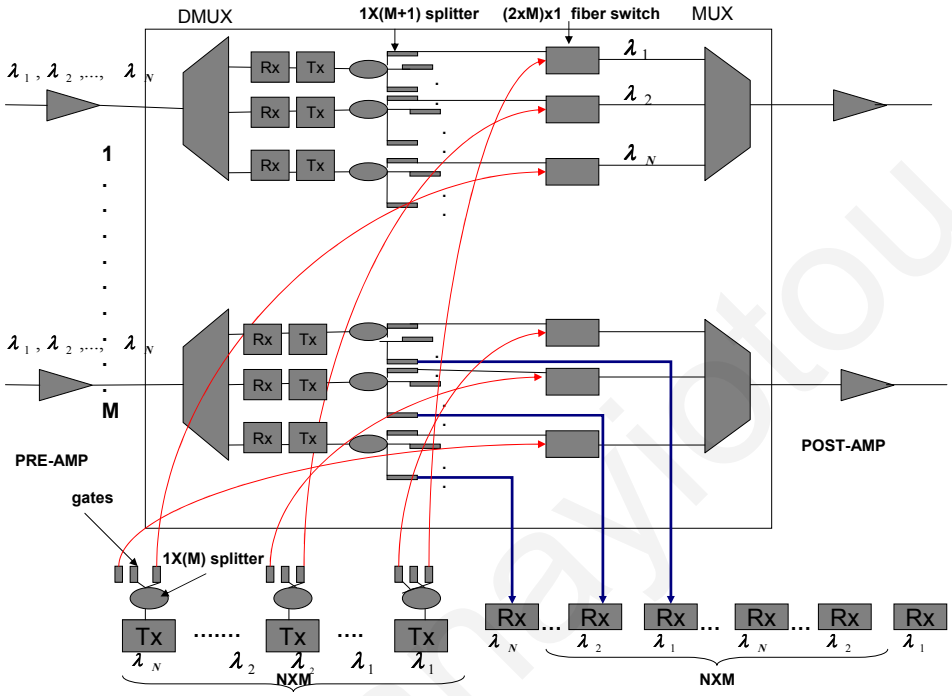

Figure 7.1: Node architecture with fixed TXs/RXs when regeneration capability is included.

For the case of regenerators placed in the network, several problems can be investigated. For example, given a fixed number of regenerators, the placement of these regenerators at specific locations while for example maximizing the network performance is a problem of interest. Another problem could be to minimize the number of regenerators required in order for every multicast request to be established into the network. Integer Linear Programming (ILP) techniques can be used to find optimal solutions to these problems. Heuristic approaches can also be used to find solutions that are close to the optimal. For example, the latter problem could be tackled by solving the static IA-MC-RWA problem for a given number of multicast requests and available wavelengths. For the routing sub-problem the Steiner tree heuristic can be used while for the wavelength assignment sub-problem the first-fit algorithm can be utilized. After the tree computation for each request, the requests can be prioritized according to a maximum-cost tree first scheme or a minimum-cost tree first scheme. Then for each tree, a wavelength assignment will be searched for. If no wavelength assignment is possible for the entire tree so that the PLIs are met, the wavelength that can serve the maximum number of destination nodes is chosen, and the destination nodes with a

Q-value that falls below the predetermined Q-threshold are identified. For each destination node where the PLIs are not met, a regenerator can be added into the path between that node and the source in such a way that now the signal is above the predetermined Q-threshold.

Another possible solution to the regenerator placement problem could be given by extending the QBPCH heuristic algorithm, where blocking probabilities even though better than other cycle-based schemes are still high. This can be done by sequentially adding regenerators into the network during the p-cycle decomposition procedure. In this way, backup paths are also considered during the regenerator placement procedure. As pointed out, in each iteration of the QBPCH heuristic, the maximum degree node m in the network is identified and a path p is created starting at node m and passing through every node in set $m(v)$ only once. Set $m(v)$ is defined as the set of nodes adjacent to m. Then the Q-factor at each node in set $m(v)$ is evaluated and the cycle closes between the last node $i \in m(v)$, added to path p with a Q-value above the predetermined Q-threshold q, and node m. The above procedure could be modified if a regenerator is added to node $i \in m(v)$. By doing so, the Q-factor at the nodes previously left out of the cycle is increased and now a cycle can be created between every node in set $m(v)$ and node m.

When groupcast connections are present in the network, a regenerator placement approach can also be developed similar to the regeneration placement scheme previously described for the static IA-MC-RWA problem where regenerators were placed in such a way in the network so that their number was minimized. Here, the static IA-GC-RWA algorithm is used for a number of given groupcast sessions and a number of available wavelengths. Groupcast sessions are prioritized according to the total cost of their pre-computed trees, and for each multicast tree in each session, the first-fit wavelength assignment algorithm is used. For each multicast tree belonging to the same session and is not entirely feasible, the wavelengths that can reach the greatest number of destination nodes are chosen. Then, the paths of the destination nodes belonging to every unfeasible multicast tree are identified, and their common nodes are extracted. Regenerators are placed at their common nodes if by doing so the Q-factor at every destination node in every multicast tree is improved. Otherwise, for the destination nodes that do not yet yield an acceptable Q-value, a path-by-path approach is followed for every destination node (i.e., the path for each destination node is examined separately and sequentially and a regenerator is placed in the network for each path that requires one).

Another area of future direction could be the improvement of the performance of groupcast connectivity by enhancing the IA-GC-RWA algorithm. This can be accomplished by

also taking into account the physical connectivity of the groupcast connections. For example, in the IA-GC-RWA algorithm presented in Chapter 6, groupcast sessions are decomposed into a set of multicast calls and then for each call, the IA-MC-RWA algorithm is used. However, the aforementioned IA-GC-RWA algorithm does not consider the fact that unlike the IA-MC-RWA problem, where only one multicast tree needs to be established at each instance, for a groupcast connection several multicast sets need to be established simultaneously into the network, in order to create a light-forest. Therefore, the problem that arises here concerns not only the wavelength assignment priority scheme that will be followed but also the multicast set priority scheme that needs to be followed, in order to either maximize the savings in terms of network resources or increase the Q-factor at the destination nodes of the trees. For example, the multicast set priority scheme could be based on the fact that between each multicast set MC_i of the same groupcast session, the only difference is that the multicast tree is initialized from a different source node s_i. Thus, each multicast set M_i could be correlated with each wavelength graph j according to the number f_i^j of available links that are outgoing from source node s_i and a list L could be created consisting of the f_i^j values in a descending order. The routing and wavelength assignment procedure followed then could be based on list L, aiming at enhancing the network performance by decreasing the blocking probability caused due to a low Q-factor. This will now the case, as a large f_i^j value increases the probability of the creation of trees that have more breadth than depth, thus improving the Q-factor at the destination nodes. Note that wavelength graph j is created by removing from the initial graph topology those arcs that are already in use by previously established light-trees on wavelength j.

A third possible avenue for future work could be the development of groupcast traffic grooming schemes that improve the network performance by efficiently utilizing the existing light-trees for each groupcast session. A possible solution to this problem can again be based on the observation that between any multicast set of the same session, every member node is the same, with the difference that a different node is used each time as the source node of the multicast connection. Thus, if a light-tree can be created for a multicast set in the session, then this light-tree can be reused by the rest of the multicast sets, depending on the remaining capacity of this light-tree and the rate of the sessions.

Yet another area for future work is the existence of optical splitters in only some of the network nodes. In this dissertation the assumption was that all the nodes are multicast capable nodes and they can split the optical signal as many times as the fanout of the node (plus one for the drop ports). A challenging area for future research is the study of multi-

cast provisioning and protection when now not all the nodes are multicast capable. Static heuristics and optimization techniques can be utilized to solve problems such as finding the minimum number of splitters required to accommodate all the connections, or finding the optimal locations for placing these splitters in the network. Dynamic provisioning and protection approaches could also be considered for the case of sparse splitting and also for the case where even when there is optical splitting this is not "full optical splitting", i.e., the signal is not split to all the outgoing links of each node. In the case of optical splitter placement, the physical topology of the network as well as the connectivity topology will place an important role in where the splitters are placed. For example, the optical splitters could be placed at the high degree nodes of the physical topology or the nodes with highly congested outgoing links in the connectivity topology. Furthermore, dynamic multicast provisioning techniques in sparse splitting networks must now also take into consideration the location of the splitters when calculating the corresponding light-trees. This could be achieved, for example, by finding the shortest path from the source node to the closest splitting point and then finding shortest paths from that point to the destinations of the multicast tree or to a second splitting point that brings us closest to the destinations, and so on.

As pointed out throughout this dissertation, only metropolitan are networks were considered during the design and engineering of the optical nodes. However, with the surge of bandwidth demand fueled by the development of bandwidth-hungry applications, there is a pressing need to maximize the capacity that can be transported by optical backbone networks in order to feed both business and residential customers. All-optical architectures seem to be the sole approach for increasing the bandwidth of backbone WDM networks, utilizing higher bit-rate wavelength channels of 40 Gbps, 100 Gbps, and beyond, in order to meet the growing traffic need. However, at such high bit rates optical signals are more susceptible to physical impairments such as chromatic dispersion, polarization mode dispersion, and nonlinear effects, resulting in shorter optical reach. Therefore, another direction for future work could be the design of all-optical backbone networks in which the impact of nonlinear effects such as four-wave mixing and cross-phase modulation must be also treated in a more specific manner.

Today, optical-bypass technology is being deployed in carrier backbone networks on a large scale. The reality, however, is that the resulting networks are not truly all-optical; all connections cannot be carried end-to-end solely in the optical domain. While a significant amount of regeneration can be eliminated through the deployment of new ultra-long-haul technology, a small amount is still required. The optical reach of long-haul systems currently

being deployed is on the order of $2,000$ to $4,000$ km. This increased reach, in conjunction with optical-bypass network elements, eliminates a significant amount of the regeneration in a backbone network. However, given that the longest connections in a North American backbone network are on the order of $8,000$ km, clearly some regeneration is still required [168]. Thus, the many benefits of transparent optical networking can be reaped by dividing a large backbone network into contained transparent domains, or islands of transparency. Each island would be fully transparent internally, but would require standard electronic interfaces to access it or to interconnect it to other islands. New directions for future work therefore arise, concerning the identification and interconnection of islands in the backbone network, the routing and wavelength assignment problem when islands of transparency are assumed, and the design and engineering of such networks.

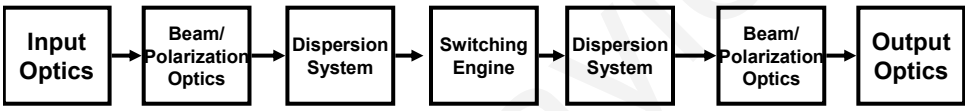

Figure 7.2: Generic WSS module implementation [194].

Finally, additional work can be performed to account for the impact of PDL/PDG on the multicast routing techniques. An efficient solution to this problem could involve utilizing wavelength selective switches (WSS) which are software-controlled fiber-optic components that select specific wavelengths from either the *Input* or the *Add* ports and route these to the *Output* ports for transmission to the next network node. In conjunction with its switching capabilities, the WSS at the same time can perform signal attenuation, thus providing channel equalization with no additional cost or complexity. A generic WSS module implementation is shown in Fig. 7.2 [194]. In such a module, optics (such as fiber and micro lens arrays) steer the light beam into/out of the device. Specialized optics then shape the collimated beam and introduce polarization diversity (utilizing prisms, cylinders, birefringent crystals, and waveplates), if needed, at the input and output of the device, followed by a dispersion system. This system can be a common grating which is polarization sensitive and thus requires incident light to be of a specific polarization. Therefore, a possible solution to the PDG/PDL problem would be to use dynamic equalization based on WSSs placed at specific points in the network. Looking at the average, best- and worst-cases, as described in Chapter 3, will be relevant then in trying to determine the placement of such switches.

Clearly, there are a number of possible avenues that one can explore as a continuation of

the work presented in this thesis. This section serves only as a guide towards some interesting future directions that could be investigated but is by no means an exhaustive list.

Bibliography

[1] G. P. Agrawal, *Fiber-Optic Communication Systems*. Wiley, 2002.

[2] M. Ali, "Optimization of splitting node placement in wavelength-routed optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1571 – 1579, Oct. 2002.

[3] M. Ali and J. Deogun, "Allocation of splitting nodes in all-optical wavelength-routed networks," *Springer Photonic Network Communications*, vol. 2, pp. 247–265, 2000.

[4] ——, "Cost-effective implementation of multicasting in wavelength-routed networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 18, no. 12, pp. 1628 –1638, Dec. 2000.

[5] ——, "Power-efficient design of multicast wavelength-routed networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, pp. 1852 – 1862, Oct. 2000.

[6] N. Antoniades, A. Boskovic, I. Tomkos, N. Madamopoulos, M. Lee, I. Roudas, D. Pastel, M. Sharma, and M. Yadlowsky, "Performance engineering and topological design of metro WDM optical networks using computer simulation," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 1, pp. 149–165, Jan. 2002.

[7] N. Antoniades, G. Ellinas, and I. E. Roudas, *WDM Systems and Networks: Modeling, Simulation, Design and Engineering*. Springer Verlag, 2012.

[8] N. Antoniades, K. Reichmann, P. Iannone, N. Frigo, A. Levine, and I. Roudas, "The impact of polarization-dependent gain on the design of cascaded semiconductor optical amplifier CWDM systems," *IEEE Photonics Technology Letters*, vol. 18, no. 20, pp. 2099–2101, Oct. 2006.

[9] N. Antoniades, K. Reichmann, P. Iannone, and A. Levine, "Engineering methodology for the use of SOAs and CWDM transmission in the metro network environment," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2006.

[10] C. Assi, A. Shami, M. Ali, Y. Ye, and S. Dixit, "Integrated routing algorithms for provisioning sub-wavelength connections in IP-over-WDM networks," *Springer Photonic Network Communications*, vol. 4, no. 3, pp. 377–390, July 2002.

[11] S. Azodolmolky, M. Klinkowski, Y. Pointurier, M. Angelou, D. Careglio, J. Sole-Pareta, and I. Tomkos, "A novel offline physical layer impairments aware RWA algorithm with dedicated path protection consideration," *IEEE/OSA Journal of Lightwave Technology*, vol. 28, no. 20, pp. 3029 –3040, Oct. 2010.

[12] S. Azodolmolky, Y. Pointurier, M. Angelou, D. Careglio, J. Sole-Pareta, and I. Tomkos, "A novel impairment aware RWA algorithm with consideration of QoT estimation inaccuracy," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 3, no. 4, pp. 290 –299, April 2011.

[13] S. Azodolmolky, Y. Pointurier, M. Klinkowski, E. Marin, D. Careglio, J. Sole-Pareta, M. Angelou, and I. Tomkos, "On the offline physical layer impairment aware RWA algorithms in transparent optical networks: State-of-the-art and beyond," in *Proc. International Conference on Optical Network Design and Modeling (ONDM)*, Braunschweig, Germany, Feb. 2009.

[14] J. Bang-Jensen and G. Gutin, *Digraphs: Theory, Algorithms and Applications, 2nd Edition*. Springer, 2009.

[15] R. Barry and S. Subramaniam, "The MAX SUM wavelength assignment algorithm for WDM ring networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Dallas, TX, Feb. 1997.

[16] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall, 1992.

[17] A. Beshir, F. Kuipers, A. Orda, and P. Van Mieghem, "Survivable impairment-aware traffic grooming in WDM rings," in *Proc. 23rd International Teletraffic Congress (ITC)*, San Francisco, CA, Sept. 2011.

[18] R. Bhandari, *Survivable Networks: Algorithms for Diverse Routing*. Kluwer Academic Publishers, 1999.

[19] K. Bharath-Kumar and J. Jaffe, "Routing to multiple destinations in computer networks," *IEEE Transactions on Communications*, vol. 31, no. 3, pp. 343–351, March 1983.

[20] A. Billah, B. Wang, and A. Awwal, "Multicast traffic grooming in WDM optical mesh networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, San Francisco, CA, Dec. 2003.

[21] A. Birman and A. Kershenbaum, "Routing and wavelength assignment methods in single-hop all-optical networks with blocking," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Boston, MA, Apr. 1995.

[22] G. Bonaventura, G. Jones, and S. Trowbridge, "Optical transport network evolution: Hot standardization topics in ITU-T including standards coordination aspects," *IEEE Communications Magazine*, vol. 46, no. 10, pp. 124–131, Oct. 2008.

[23] P. Bonenfant and A. Rodriguez-Moral, "Optical data networking," *IEEE Communications Magazine*, vol. 38, no. 3, pp. 63–70, March 2000.

[24] E. Bouillet, G. Ellinas, J.-F. Labourdette, and R. Ramamurthy, *Path Routing in Mesh Optical Networks*. Wiley-Interscience, 2007.

[25] E. Bouillet, J.-F. Labourdette, G. Ellinas, R. Ramamurthy, and S. Chaudhuri, "Stochastic approaches to compute shared mesh restored lightpaths in optical network architectures," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, New York, NY, 2002.

[26] E. Bouillet, J.-F. Labourdette, R. Ramamurthy, and S. Chaudhuri, "Enhanced algorithm cost model to control tradeoffs in provisioning shared mesh restored lightpaths," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2002.

[27] M. Bourouha, M. Bataineh, and M. Guizani, "Advances in optical switching and networking: Past, present, and future," in *Proc. IEEE SoutheastCon*, Columbia, South Carolina, Apr. 2002.

[28] Y. Cao and O. Yu, "On the study of group multicast in WDM networks," in *Proc. IEEE International Conference on Communications (ICC)*, Seoul, Korea, May 2005.

[29] ——, "Optimal placement of light splitters and wavelength converters for multicast in WDM networks," in *Proc. International Conference on Communications, Circuits and Systems (ICCCAS)*, Hong Kong, China, May 2005.

[30] ——, "QoS-guaranteed routing and wavelength assignment for group multicast in optical WDM networks," in *Proc. International Conference on Optical Network Design and Modeling (ONDM)*, Milan, Italy, Feb. 2005.

[31] ——, "Groupcast in wavelength-routed WDM networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 24, no. 11, pp. 4286 –4295, Nov. 2006.

[32] R. Cardillo, V. Curri, and M. Mellia, "Considering transmission impairments in wavelength routed networks," in *Proc. International Conference on Optical Network Design and Modeling (ONDM)*, Milan, Italy, Feb. 2005.

[33] ——, "Considering transmission impairments in configuring wavelength routed optical networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2006.

[34] T. Carpenter, R. Menendez, D. Shallcross, J. Gannett, J. Jackel, and A. Von Lehmen, "Cost-conscious impairment-aware routing," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Los Angeles, CA, Feb. 2004.

[35] A. Carter, "Evolution of optical component technologies for access and metro networks," in *Proc. 35th European Conference on Optical Communication (ECOC)*, Vienna, Austria, Sep. 2009.

[36] B. Chen and J. Wang, "Efficient routing and wavelength assignment for multicast in WDM networks," *IEEE Journal on Selected Areas in Communications,*, vol. 20, no. 1, pp. 97–109, Jan. 2002.

[37] B. Chen, G. Rouskas, and R. Dutta, "Clustering methods for hierarchical traffic grooming in large-scale mesh WDM networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 2, no. 8, pp. 502 –514, Aug. 2010.

[38] A. Chiu, G. Li, and D.-M. Hwang, "New problems on wavelength assignment in ULH networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2006.

[39] I. Chlamtac, A. Ganz, and G. Karmi, "Purely optical networks for terabit communication," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Ottawa, Canada, Apr. 1989.

[40] ——, "Lightnet: Lightpath based solutions for wide bandwidth WANs," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, San Francisco, CA, June 1990.

[41] S. Cho and T. Lee, "Minimum cost multicast routing based on high utilization MC nodes suited to sparse-splitting optical networks," in *Proc. International Conference on Computational Science and its Applications*, Glasgow, UK, May 2006.

[42] J. S. Choi, N. Golmie, F. Lapeyrere, F. Mouveaux, and D. Su, "A functional classification of routing and wavelength assignment schemes in DWDM networks: Static case," in *Proc. International Conference on Optical Communication and Networks (OPNET)*, Jan. 2000.

[43] G. Chowdhary and C. Murthy, "Dynamic multicast traffic engineering in WDM groomed mesh networks," in *Proc. 1st International Conference on Broadband Communications, Networks, and Systems (BROADNETS)*, San Jose, CA, 2004.

[44] T. Cormen, *Introduction to algorithms*. The MIT press, 2001.

[45] F. Cugini, N. Andriolli, L. Valcarenghi, and P. Castoldi, "A novel signaling approach to encompass physical impairments in GMPLS networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Dallas, TX, Nov. 2004.

[46] F. Curti, B. Daino, G. de Marchis, and F. Matera, "Statistical treatment of the evolution of the principal states of polarization in single-mode fibers," *IEEE/OSA Journal Lightwave Technology*, vol. 2, no. 14, pp. 1162–1166, August 1990.

[47] T. De and S. Sen, "Multicast routing and wavelength assignment in sparse splitting all optical networks," in *Proc. IFIP International Conference on Wireless and Optical Communications Networks*, Bangalore, India, Apr. 2006.

[48] J. de Santi, A. Drummond, N. da Fonseca, X. Chen, and A. Jukan, "Leveraging multipath routing and traffic grooming for an efficient load balancing in optical networks," in *Proc. IEEE International Conference on Communications (ICC)*, Budapest, Hungary, June 2012.

[49] T. Deng and S. Subramaniam, "Adaptive QoS routing in dynamic wavelength-routed optical networks," in *Proc. 2nd International Conference on Broadband Communications, Networks and Systems (BROADNETS)*, Boston, MA, Oct. 2005.

[50] D. Derrichkson, *Fiber optics test and measurement*. Prentice Hall, Upper Saddle River, 1998.

[51] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numeriche Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[52] A. Ding and G.-S. Poo, "A survey of optical multicast over WDM networks," *Elsevier Computer Communications*, vol. 26, no. 2, pp. 193 – 200, Feb. 2003.

[53] B. T. Doshi, S. Dravida, P. Harshavardhana, O. Hauser, and Y. Wang, "Optical network design and restoration," *Bell Labs Technical Journal*, vol. 4, no. 1, pp. 58 – 84, January-March 1999.

[54] J. Downie and A. Ruffin, "Analysis of signal distortion and crosstalk penalties induced by optical filters in optical networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, no. 9, pp. 1876–1886, Sept. 2003.

[55] R. Dutta and G. Rouskas, "Traffic grooming in WDM networks: Past and future," *IEEE Network*, vol. 16, no. 6, pp. 46–56, Nov. 2002.

[56] G. Ellinas, N. Antoniades, T. Panayiotou, A. Hadjiantonis, and A. Levine, "Multicast routing algorithms based on Q-factor physical-layer constraints in metro networks," *IEEE Photonics Technology Letters*, vol. 21, no. 6, pp. 365–367, March 2009.

[57] G. Ellinas, E. Bouillet, R. Ramamurthy, J.-F. Labourdette, S. Chaudhuri, and K. Bala, "Routing and restoration architectures in mesh optical networks," *SPIE Optical Networks Magazine*, vol. 4, no. 1, pp. 91 – 106, January/February 2003.

[58] G. Ellinas, A. Hailemariam, and T. Stern, "Protection cycles in mesh WDM networks," *IEEE Journal on Selected Areas in Communications,*, vol. 18, no. 10, pp. 1924 – 1937, Oct. 2000.

[59] S. Even, A. Itai, and A. Shamir, "On the complexity of time table and multicommodity flow problems," in *Proc. 16th Annual Symposium on Foundations of Computer Science (FOCS)*, Berkeley, CA, Oct. 1975.

[60] M. Ezzahdi, S. Al Zahr, M. Koubaa, N. Puech, and M. Gagnaire, "LERP: A quality of transmission dependent heuristic for routing and wavelength assignment in hybrid WDM networks," in *Proc. 15th International Conference on Computer Communications and Networks (ICCCN)*, Arlington, Virginia, Oct. 2006.

[61] A. Fei, J. Cui, M. Gerla, and D. Cavendish, "A dual-tree scheme for fault-tolerant multicast," in *Proc. IEEE International Conference on Communications (ICC)*, Helsinki, Finland, June 2001.

[62] 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)*, Beijing, China, May 2008.

[63] C. Gao, H. Cankaya, A. Patel, J. Jue, X. Wang, Q. Zhang, P. Palacharla, and M. Sekiya, "Survivable impairment-aware traffic grooming and regenerator placement with connection-level protection," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 4, no. 3, pp. 259 –270, March 2012.

[64] L. Gillner, "Transmission limitations in the all-optical network," in *Proc. 22nd European Conference on Optical Communication (ECOC)*, Oslo, Norway, Sep. 1996.

[65] W. Grover, *Mesh-based Survivable Transport Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking*. Upper Saddle River, New Jersey: Prentice Hall, Aug. 2003.

[66] W. 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)*, Atlanta, GA, June 1998.

[67] A. Hamad and A. Kamal, "Optimal power-aware design of all-optical multicasting in wavelength routed networks," in *Proc. IEEE International Conference on Communications (ICC)*, Paris, France, June 2004.

[68] ——, "Routing and wavelength assignment with power aware multicasting in WDM networks," in *Proc. 2nd International Conference on Broadband Communications, Networks and Systems (BROADNETS)*, Boston, MA, Oct. 2005.

[69] A. M. Hamad and A. E. Kamal, "Power-aware connection provisioning for all-optical multicast traffic in WDM networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 2, no. 7, pp. 481–495, July 2010.

[70] J. He, M. Brandt-Pearce, Y. Pointurier, C. Brown, and S. Subramaniam, "Adaptive wavelength assignment using wavelength spectrum separation for distributed optical networks," in *Proc. IEEE International Conference on Communications (ICC)*, Glasgow, Scotland, June 2007.

[71] J. He, M. Brandt-Pearce, Y. Pointurier, and S. Subramaniam, "QoT-aware routing in impairment-constrained optical networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Washington, DC, Nov. 2007.

[72] J. He, M. Brandt-Pearce, and S. Subramaniam, "QoS-aware wavelength assignment with BER and latency guarantees for crosstalk limited networks," in *Proc. IEEE International Conference on Communications (ICC)*, Glasgow, Scotland, June 2007.

[73] J. He, S.-H. Chan, and D. Tsang, "Routing and wavelength assignment for WDM multicast networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, San Antonio, TX, Nov. 2001.

[74] P.-H. Ho, "State-of-the-art progress in developing survivable routing schemes in mesh WDM networks," *IEEE Communications Surveys and Tutorials*, vol. 6, no. 4, pp. 2–16, 2004.

[75] P.-H. Ho and H. Mouftah, "A framework for service-guaranteed shared protection in WDM mesh networks," *IEEE Communications Magazine*, vol. 40, no. 2, pp. 97–103, Feb. 2002.

[76] D. S. Hochbaum, *Approximation algorithms for NP-hard problems*. PWS Publishing Co., 1997.

[77] C.-Y. Hsieh and W. Liao, "All-optical multicast routing in sparse splitting WDM networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 6, pp. 51–62, Aug. 2007.

[78] Y. Huang, J. Heritage, and B. Mukherjee, "Connection provisioning with transmission impairment consideration in optical WDM networks with high-speed channels," *IEEE/OSA Journal of Lightwave Technology*, vol. 23, no. 3, pp. 982 – 993, March 2005.

[79] Z. Hui, C. Ou, and B. Mukherjee, "Path-protection routing and wavelength assignment (RWA) in WDM mesh networks under duct-layer constraints," *IEEE/ACM Transactions on Networking*, vol. 11, no. 2, pp. 248 – 258, Apr. 2003.

[80] G. Jeong and E. Ayanoglu, "Comparison of wavelength-interchanging and wavelength-selective cross-connects in multiwavelength all-optical networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, San Francisco, CA, March 1996.

[81] A. Jukan and G. Franzl, "Constraint-based path selection methods for on-demand provisioning in WDM networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, New York, NY, June 2002.

[82] A. Kamal and R. Ul-Mustafa, "Multicast traffic grooming in WDM networks," in *Proc. SPIE/IEEE Fourth Annual Optical Networking and Communications Conference (Opticomm)*, Dallas, TX, Oct. 2003.

[83] A. Kamal, "Algorithms for multicast traffic grooming in WDM mesh networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp. 96–105, Nov. 2006.

[84] T. Kamalakis, T. Sphicopoulos, and M. Sagriotis, "Accurate estimation of the error probability in the presence of in-band crosstalk noise in WDM networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, no. 10, pp. 2172 – 2181, Oct. 2003.

[85] E. Karasan and E. Ayanoglu, "Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks," *IEEE/ACM Transactions on Networking*, vol. 6, no. 2, pp. 186–196, Apr. 1998.

[86] S. Kartakopoulos, *Introduction to DWDM Technology: Data in a Rainbow*. SPIE Optical Engineering Press, 2000.

[87] A. Khalil, A. Hadjiantonis, C. Assi, A. Shami, G. Ellinas, and M. Ali, "Dynamic provisioning of low-speed unicast/multicast traffic demands in mesh-based WDM optical networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 24, no. 2, pp. 681–693, Feb. 2006.

[88] A. Khalil, A. Hadjiantonis, G. Ellinas, and M. Ali, "Dynamic provisioning of survivable heterogeneous multicast and unicast traffic in WDM networks," in *Proc. IEEE International Conference on Communications (ICC)*, Istanbul, Turkey, June 2006.

[89] ——, "Pre-planned multicast protection approaches in WDM mesh networks," in *Proc. 31st European Conference on Optical Communication (ECOC)*, Glasgow, Scotland, Sep. 2005.

[90] M. Kiaei, C. Assi, and B. Jaumard, "A survey on the p-cycle protection method," *IEEE Communications Surveys Tutorials*, vol. 11, no. 3, pp. 53 –70, 2009.

[91] T. Koch and J. Bowers, "Nature of wavelength chirping in directly modulated semiconductor lasers," *Electronics Letters*, vol. 20, no. 25, pp. 1038–1040, Dec. 1984.

[92] P. Kulkarni, A. Tzanakaki, C. Machuka, and I. Tomkos, "Benefits of Q-factor based routing in WDM metro networks," in *Proc. 31st European Conference on Optical Communication (ECOC)*, Glasgow, Scotland, Sep. 2005.

[93] K. Lee and M. Shayman, "Optical network design with optical constraints in multihop WDM mesh networks," in *Proc. 13th International Conference on Computer Communications and Networks (ICCCN)*, Chicago, IL, Oct. 2004.

[94] M. Lee, N. Antoniades, and A. Boskovic, "PDL-induced channel power divergence in a metro WDM network," *IEEE Photonics Technology Letters*, vol. 14, no. 4, pp. 561 –563, Apr. 2002.

[95] W. Liang and Y. Liu, "Online broadcasting and multicasting in WDM networks with shared light splitter bank," in *Proc. Fourth International Conference on Broadband Communications, Networks and Systems (BROADNETS)*, Raleigh, NC, Sep. 2007.

[96] L. Liao, L. Li, and S. Wang, "Multicast protection scheme in survivable WDM optical networks," *Elsevier Journal of Network and Computer Applications*, vol. 31, no. 3, pp. 303 – 316, August 2008.

[97] M. Lima, A. Cesar, and A. Araujo, "Optical network optimization with transmission impairments based on genetic algorithm," in *Proc. SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, Foz do Iguacu, Brazil, Sep. 2003.

[98] R. Lin, W.-D. Zhong, S. K. Bose, and M. Zukerman, "Leaking strategy for multicast traffic grooming in WDM mesh networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 30, no. 23, pp. 3709 –3719, Dec. 2012.

[99] R. Lin, W.-D. Zhong, S. Bose, and M. Zukerman, "Design of WDM networks with multicast traffic grooming," *IEEE/OSA Journal of Lightwave Technology*, vol. 29, no. 16, pp. 2337 –2349, Aug. 2011.

[100] ——, "Heuristic algorithms for multicast traffic grooming in WDM mesh networks," in *Proc. 8th International Conference on Information, Communications and Signal Processing (ICICS)*, Singapore, Dec. 2011.

[101] ——, "Multicast traffic grooming in tap-and-continue WDM mesh networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 4, no. 11, pp. 918 –935, Nov. 2012.

[102] R. Lin, W.-D. Zhong, S. Bose, M. Zukerman, and Q. Huang, "Light-tree based multicast traffic grooming in WDM mesh networks," in *Proc. 15th Optoelectronics and Communications Conference (OECC)*, Sapporo, Japan, July 2010.

[103] K. H. Liu, *IP over WDM*. Wiley, 2002.

[104] L. Long and A. Kamal, "Tree-based protection of multicast services in WDM mesh networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Honolulu, Hawaii, Dec. 2009.

[105] C. Lu, H. Luo, S. Wang, and L. Li, "A novel shared segment protection algorithm for multicast sessions in mesh WDM networks," *ETRI Journal*, vol. 28, no. 3, pp. 329–336, June 2006.

[106] S. Lumetta, M. Medard, and Y.-C. Tseng, "Capacity versus robustness: A tradeoff for link restoration in mesh networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 18, no. 12, pp. 1765 –1775, Dec. 2000.

[107] H. Luo, L. Li, and H. Yu, "Algorithm for protecting light-trees in survivable mesh wavelength-division-multiplexing networks," *OSA Journal of Optical Networking*, vol. 5, no. 12, pp. 1071–1083, December 2006.

[108] H. Luo, L. Li, H. Yu, and S. Wang, "Achieving shared protection for dynamic multicast sessions in survivable mesh WDM networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 9, pp. 83 –95, Dec. 2007.

[109] M. H. Macgregor, W. D. Grover, and K. Ryhorchuk, "Optimal spare capacity preconfiguration for faster restoration of mesh networks," *Springer Journal of Network and Systems Management*, vol. 5, no. 2, pp. 159–171, June 1997.

[110] N. Madamopoulos, D. Friedman, I. Tomkos, and A. Boskovic, "Study of the performance of a transparent and reconfigurable metropolitan area network," *IEEE/OSA Journal of Lightwave Technology*, vol. 20, no. 6, pp. 937–945, June 2002.

[111] G. Maier, A. Pattavina, S. De Patre, and M. Martinelli, "Optical network survivability: Protection techniques in the WDM layer," *Springer Photonic Network Communications*, vol. 4, pp. 251–269, July 2002.

[112] R. Malli, X. Zhang, and C. Qiao, "Benefits of multicasting in all-optical networks," in *Proc. of SPIE Conference On All-optical Networking*, vol. 3531, Boston, MA, 1998, pp. 209–220.

[113] E. Marin, S. Sanchez, X. Masip, J. Sole, G. Maier, W. Erangoli, S. Santoni, and M. Quagliotti, "Applying prediction concepts to routing on semi-transparent optical transport networks," in *Proc. 9th International Conference on Transparent Optical Networks (ICTON)*, Rome, Italy, July 2007.

[114] G. Markidis, S. Sygletos, A. Tzanakaki, and I. Tomkos, "Impairment aware based routing and wavelength assignment in transparent long haul networks," in *Proc. International Conference on Optical Network Design and Modeling (ONDM)*, Athens, Greece, May 2007.

[115] ——, "Impairment-constraint-based routing in ultralong-haul optical networks with 2R regeneration," *IEEE Photonics Technology Letters*, vol. 19, no. 6, pp. 420–422, March 2007.

[116] A. Marsden, A. Maruta, and K.-I. Kitayama, "Routing and wavelength assignment encompassing FWM in WDM lightpath networks," in *Proc. International Conference on Optical Network Design and Modeling (ONDM)*, Vilanova i la Geltru, Spain, March 2008.

[117] J. Martins-Filho, C. Bastos-Filho, E. Arantes, S. Oliveira, L. Coelho, J. de Oliveira, R. Dante, E. Fontana, and F. Nunes, "Novel routing algorithm for transparent optical networks based on noise figure and amplifier saturation," in *Proc. SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, Sep. 2003.

[118] M. Medard, R. Barry, S. Finn, W. He, and S. Lumetta, "Generalized loop-back recovery in optical mesh networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 1, pp. 153 –164, Feb. 2002.

[119] E. Modiano, "Traffic grooming in WDM networks," *IEEE Communications Magazine*, vol. 39, no. 7, pp. 124 –129, July 2001.

[120] D. Monoyios and K. Vlachos, "Multiobjective genetic algorithms for solving the impairment-aware routing and wavelength assignment problem," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 3, no. 1, pp. 40 –47, Jan. 2011.

[121] A. Morea, N. Brogard, F. Leplingard, J.-C. Antona, T. Zami, B. Lavigne, and D. Bayart, "QoT function and a* routing: An optimized combination for connection search in translucent networks," *OSA Journal of Optical Networking*, vol. 7, no. 1, pp. 42–61, Jan. 2008.

[122] B. Mukherjee, Y. Huang, and J. Heritage, "Impairment-aware routing in wavelength-routed optical networks," in *Proc. 17th Annual Meeting of the IEEE Lasers and Electro-Optics Society (LEOS)*, Rio Grande, Puerto Rico, Nov. 2004.

[123] B. Mukherjee, *Optical WDM Networks*. Springer, 2006.

[124] A. Nag, M. Tornatore, and B. Mukherjee, "Optical network design with mixed line rates and multiple modulation formats," *IEEE/OSA Journal of Lightwave Technology*, vol. 28, no. 4, pp. 466 –475, Feb. 2010.

[125] D. Onguetou and W. Grover, "p-Cycle network design: From fewest in number to smallest in size," in *Proc. 6th International Workshop on Design and Reliable Communication Networks (DRCN)*, La Rochelle, France, Oct. 2007.

[126] S. Pachnicke, T. Paschenda, and P. Krummrich, "Assessment of a constraint-based routing algorithm for translucent 10Gbits/s DWDM networks considering fiber nonlinearities," *OSA Journal of Optical Networking*, vol. 7, no. 4, pp. 365–377, Apr. 2008.

[127] ——, "Physical impairment based regenerator placement and routing in translucent optical networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, San Diego, CA, Feb. 2008.

[128] T. Panayiotou, G. Ellinas, N. Antoniades, and A. Hadjiantonis, "Node architecture design and network engineering impact on optical multicasting based on physical layer constraints," in *Proc. 12th International Conference on Transparent Optical Networks (ICTON)*, Munich, Germany, June 2010.

[129] ——, "A novel segment-based protection algorithm for multicast sessions in optical networks with mesh topologies," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Los Angeles, CA, March 2011.

[130] T. Panayiotou, G. Ellinas, N. Antoniades, and A. Levine, "Designing and engineering metropolitan area transparent optical networks for the provisioning of multicast sessions," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, San Diego, CA, March 2010.

[131] R. Pankaj, "Wavelength requirements for multicasting in all-optical networks," *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 414–424, June 1999.

[132] G. Pavani and H. Waldman, "Adaptive routing and wavelength assignment with power constraints using ant colony optimization," in *Proc. International Telecommunications Symposium (ITS)*, Fortaleza, Ceara, Brazil, Sep. 2006.

[133] G. Pavani, L. Zuliani, H. Waldman, and M. Magalhes, "Distributed approaches for impairment-aware routing and wavelength assignment algorithms in GMPLS networks," *Elsevier Computer Networks*, vol. 52, no. 10, pp. 1905 – 1915, July 2008.

[134] H. Pereira, D. Chaves, C. Bastos-Filho, and J. Martins-Filho, "Impact of physical layer impairments in all-optical networks," in *Proc. SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, Salvador, Brazil, Nov. 2007.

[135] Y. Pointurier, M. Brandt-Pearce, T. Deng, and S. Subramaniam, "Fair routing and wavelength assignment in all-optical networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2006.

[136] C. Politi, V. Anagnostopoulos, C. Matrakidis, and A. Stavdas, "Physical layer impairment aware routing algorithms based on analytically calculated Q-factor," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2006.

[137] G.-S. Poo and Y. Zhou, "A new multicast wavelength assignment algorithm in wavelength-routed WDM networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 4, pp. 2–12, April 2006.

[138] A. Rahbar and O. Yang, "Contention avoidance and resolution schemes in bufferless all-optical packet-switched networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 10, no. 4, pp. 94–107, 2008.

[139] Q. Rahman, S. Bandyopadhyay, and Y. Aneja, "A branch, price and cut approach for optimal traffic grooming in WDM optical networks," in *Proc. IEEE International Conference on Communications (ICC)*, Kyoto, Japan, June 2011.

[140] T. Rahman, M. Ali, and G. Ellinas, "Building light-forest to support group multicast in mesh-based optical grid networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2006.

[141] T. Rahman and G. Ellinas, "Protection of multicast sessions in WDM mesh optical networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2005.

[142] T. Rahman, G. Ellinas, and M. Ali, "Lightpath- and light-tree-based groupcast routing and wavelength assignment in mesh optical networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 1, no. 2, pp. A44 –A55, July 2009.

[143] ——, "Performance evaluation of light-forests to serve groupcast sessions in WDM mesh networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, San Diego, CA, March 2009.

[144] B. Ramamurthy, D. Datta, H. Feng, J. Heritage, and B. Mukherjee, "Impact of transmission impairments on the teletraffic performance of wavelength-routed optical networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 17, no. 10, pp. 1713 –1723, Oct. 1999.

[145] B. Ramamurthy, G. N. Rouskas, and K. M. Sivalingam, *Next-Generation Internet: Architectures and Protocols*. Cambridge University Press, 2011.

[146] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, no. 4, pp. 870 – 883, Apr. 2003.

[147] ——, "Survivable WDM mesh networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, no. 4, pp. 870–883, Apr. 2003.

[148] R. Ramaswami and G. Sasaki, "Multiwavelength optical networks with limited wavelength conversion," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Kobe, Japan, Apr. 1997.

[149] R. Ramaswami, K. Sivarajan, and G. Sasaki, *Optical Networks: A Practical Perspective*. Morgan Kaufmann, 2010.

[150] R. Ramaswami and K. Sivarajan, "Optimal routing and wavelength assignment in all-optical networks," in *Proc. 13th IEEE International Conference on Computer Communications (INFOCOM)*, Toronto, Canada, June 1994.

[151] O. Rofidal, L. Berthelon, O. Audouin, A. Bisson, L. Noirie, C. Drion, N. Collin, P. Bonno, J. Chauvin, and F. Rayemaekers, "Design of a Pan-European lightwave core and access networking trial (ACTS-PELICAN project)," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Baltimore, MD, March 2000.

[152] I. Roudas, N. Antoniades, T. Otani, T. Stern, R. Wagner, and D. Chowdhury, "Accurate modeling of optical multiplexer/demultiplexer concatenation in transparent multiwavelength optical networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 20, no. 6, pp. 921 –936, June 2002.

[153] A. Rubio-Largo, M. Vega-Rodriguez, J. Gomez-Pulido, and J. Sanchez-Perez, "Multiobjective metaheuristics for traffic grooming in optical networks," *IEEE Transactions on Evolutionary Computation*, vol. PP, no. 99, pp. 1–17, June 2012.

[154] R. Sabella, E. Iannone, M. Listanti, M. Berdusco, and S. Binetti, "Impact of transmission performance on path routing in all-optical transport networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 16, no. 11, pp. 1965 –1972, Nov. 1998.

[155] L. Sahasrabuddhe and B. Mukherjee, "Light trees: Optical multicasting for improved performance in wavelength routed networks," *IEEE Communications Magazine*, vol. 37, no. 2, pp. 67–73, Feb. 1999.

[156] ——, "Multicast routing algorithms and protocols: A tutorial," *IEEE Network*, vol. 14, no. 1, pp. 90–102, Jan. 2000.

[157] G. Sahin and M. Azizoglu, "Multicast routing and wavelength assignment in wide-area networks," in *Proc. of SPIE Conference On All-optical Networking*, vol. 3531, Boston, MA, Oct. 1998, pp. 196–208.

[158] A. Saleh, "Transparent optical networking in backbone networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, vol. 3, Baltimore, 2000.

[159] M. Saleh and A. Kamal, "Design and provisioning of WDM networks for many-to-many traffic grooming," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Honolulu, Hawaii, Dec. 2009.

[160] ——, "Design and provisioning of WDM networks with many-to-many traffic grooming," *IEEE/ACM Transactions on Networking*, vol. 18, no. 6, pp. 1869 –1882, Dec. 2010.

[161] ——, "Approximation algorithms for many-to-many traffic grooming in optical WDM networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 5, pp. 1527 –1540, Oct. 2012.

[162] E. Salvadori, Y. Ye, A. Zanardi, H. Woesner, M. Carcagni, G. Galimberti, G. Martinelli, A. Tanzi, and D. La Fauci, "Signaling-based architectures for impairment-aware lightpath set-up in GMPLS networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Washington, DC, Nov. 2007.

[163] E. Salvadori, Y. Ye, A. Zanardi, H. Woesner, M. Carcagnì, G. Galimberti, G. Martinelli, A. Tanzi, and D. La Fauci, "A study of connection management approaches for an impairment-aware optical control plane," in *Proc. International Conference on Optical Network Design and Modeling (ONDM)*, Athens, Greece, 2007.

[164] N. Sambo, N. Andriolli, A. Giorgetti, L. Valcarenghi, I. Cerutti, P. Castoldi, and F. Cugini, "GMPLS-controlled dynamic translucent optical networks," *IEEE Network*, vol. 23, no. 3, pp. 34–40, May 2009.

[165] C. Saradhi and S. Subramaniam, "Physical layer impairment aware routing (PLIAR) in WDM optical networks: Issues and challenges," *IEEE Communications Surveys Tutorials*, vol. 11, no. 4, pp. 109–130, Dec. 2009.

[166] K. Shrikhande, I. White, D. Wonglumsom, S. Gemelos, M. Rogge, Y. Fukashiro, M. Avenarius, and L. Kazovsky, "HORNET: A packet-over-WDM multiple access metropolitan area ring network," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, pp. 2004 –2016, Oct. 2000.

[167] J. Simmons, "On determining the optimal optical reach for a long-haul network," *IEEE/OSA Journal of Lightwave Technology*, vol. 23, no. 3, pp. 1039–1048, March 2005.

[168] ——, "Network design in realistic "all-optical" backbone networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp. 88 –94, Nov. 2006.

[169] N. Singhal and B. Mukherjee, "Architectures and algorithm for multicasting in WDM optical mesh networks using opaque and transparent optical cross-connects," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2001.

[170] ——, "Algorithms for provisioning survivable multicast sessions against link failures in mesh networks," in *Proc. 5th International Workshop of Distributed Computing (IWDC)*, Kolkata, India, Dec. 2003.

[171] ——, "Protecting multicast sessions in WDM optical mesh networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, no. 4, pp. 884 – 892, Apr. 2003.

[172] N. Singhal, C. Ou, and B. Mukherjee, "Cross-sharing vs. self-sharing trees for protecting multicast sessions in mesh networks," *Elsevier Computer Networks*, vol. 50, no. 2, pp. 200–206, February 2006.

[173] N. Singhal, L. Sahasrabuddhe, and B. Mukherjee, "Protecting a multicast session against single link failures in a mesh network," in *Proc. IEEE International Conference on Communications (ICC)*, Anchorage, Alaska, May 2003.

[174] ——, "Provisioning of survivable multicast sessions against single link failures in optical WDM mesh networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, no. 11, pp. 2587–2594, Nov. 2003.

[175] J. E. Smith, "A study of branch prediction strategies," in *Proc. 8th Annual Symposium on Computer Architecture (ISCA)*, Minneapolis, MN, May 1981.

[176] P. Soproni and T. Cinkler, "Physical impairment aware multicast routing heuristics," in *Proc. 13th International Conference on Transparent Optical Networks (ICTON)*, Stockholm, Sweden, June 2011.

[177] N. Sreenath, N. Krishna Mohan Reddy, G. Mohan, and C. Siva Ramamurthy, "Virtual source based multicast routing in WDM networks with sparse light splitting," in *Proc. IEEE Workshop on High Performance Switching and Routing (HPSR)*, Dallas, TX, May 2001.

[178] N. Sreenath, K. Satheesh, G. Mohan, and C. Murthy, "Virtual source based multicast routing in WDM optical networks," in *Proc. IEEE International Conference on Networks (ICON)*, Gdansk, Poland, June 2000.

[179] D. Stamatelakis, *Theory and Algorithms for Preconfiguration of Spare Capacity in Mesh Restorable Networks*. Master Thesis, University of Alberta, 1997.

[180] D. Stamatelakis and W. Grover, "Theoretical underpinnings for the efficiency of restorable networks using preconfigured cycles," *IEEE Transactions on Communications*, vol. 48, no. 8, pp. 1262 –1265, Aug. 2000.

[181] T. E. Stern and K. Bala, *Multiwavelength Optical Networks: A Layered Approach*. Prentice Hall, 1999.

[182] T. E. Stern, G. Ellinas, and K. Bala, *Multiwavelength Optical Networks: Architectures, Design, and Control*. Cambridge University Press, 2008.

[183] S. Subramaniam and R. Barry, "Wavelength assignment in fixed routing WDM networks," in *Proc. IEEE International Conference on Communications (ICC)*, Montreal, Canada, June 1997.

[184] Y. Sun, J. Gu, and D. H. K. Tsang, "Routing and wavelength assignment in all optical networks with multicast traffic," in *Proc. ITC-16*, Edinburgh, Scotland, June 1999.

[185] ——, "Multicast routing in all-optical wavelength-routed networks," *SPIE Optical Networks Magazine*, vol. 2, no. 4, pp. 101–109, Aug. 2001.

[186] J. Suurballe, "Disjoint paths in a network," *Networks*, vol. 4, no. 2, pp. 125–145, 1974.

[187] H. Takahashi and A. Matsuyama, "An approximate solution for the Steiner problem in graphs," *Math. Japonica*, vol. 24, no. 6, pp. 573–577, 1980.

[188] S. Thiagarajan and A. Somani, "Capacity fairness of WDM networks with grooming capabilities," in *Proc. SPIE Optical Networking and Communications Conference (Opticomm)*, vol. 4233, Dallas, TX, Oct. 2000, pp. 191–201.

[189] I. Tomkos, D. Vogiatzis, C. Mas, I. Zacharopoulos, A. Tzanakaki, and E. Varvarigos, "Performance engineering of metropolitan area optical networks through impairment constraint routing," *IEEE Communications Magazine*, vol. 42, no. 8, pp. S40–S47, Aug. 2004.

[190] T. Tripathi and K. Sivarajan, "Computing approximate blocking probabilities in wavelength routed all-optical networks with limited-range wavelength conversion," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, pp. 2123–2129, Oct. 2000.

[191] W.-Y. Tseng and S.-Y. Kuo, "All-optical multicasting on wavelength-routed WDM networks with partial replication," in *Proc. 15th International Conference on Information Networking (ICOIN)*, Beppu City, Oita, Japan, Jan. 2001.

[192] R. Ul-Mustafa and A. Kamal, "Design and provisioning of WDM networks with multicast traffic grooming," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 4, pp. –53, Apr. 2006.

[193] B. Vandegriend, *Finding Hamiltonian Cycles: Algorithms, Graphs and Performance*. Master Thesis, University of Alberta, 1998.

[194] P. Wall, P. Colbourne, C. Reimer, and S. McLaughlin, "WSS switching engine technologies," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, San Diego, CA, Feb. 2008.

[195] X. Wan, Y. Li, H. Zhang, and X. Zheng, "Dynamic traffic grooming in flexible multilayer IP/optical networks," *IEEE Communications Letters*, vol. 16, no. 12, pp. 2079 –2082, Dec. 2012.

[196] J. Wang, B. Chen, and R. Uma, "Dynamic wavelength assignment for multicast in all-optical WDM networks to maximize the network capacity," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 8, pp. 1274 – 1284, Oct. 2003.

[197] S.-W. Wang and C.-Y. Wen, "Lightpath-level active rerouting algorithms in all-optical WDM networks with alternate routing and traffic grooming," in *Proc. International Conference on Information Networking (ICOIN)*, Bangkok, Thailand, Feb. 2012.

[198] X. Wang, L. Guo, L. Pang, J. Du, and F. Jin, "Segment protection algorithm with load balancing for multicasting WDM mesh networks," in *Proc. 10th International Conference on Advanced Communication Technology (ICACT)*, Phoenix Park, Korea, Feb. 2008.

[199] H. Wanjun, T. Limin, M. Razo, A. Sivasankaran, M. Tacca, and A. Fumagalli, "Routing and wavelength assignment computed jointly for a given set of multicast trees reduces the total wavelength conversion," in *Proc. 12th International Conference on Transparent Optical Networks (ICTON)*, Munich, Germany, June 2010.

[200] C.-S. Wu, S.-W. Lee, and Y.-T. Hou, "Backup VP preplanning strategies for survivable multicast ATM networks," in *Proc. IEEE International Conference on Communications (ICC)*, Montreal, Canada, June 1997.

[201] K.-D. Wu, J.-C. Wu, and C.-S. Yang, "Multicast routing with power consideration in sparse splitting WDM networks," in *Proc. IEEE International Conference on Communications (ICC)*, Helsinki, Finland, June 2001.

[202] T.-H. Wu, *Fiber Network Service Survivability*. Norwood, MA: Inc. Artech House, 1992.

[203] Y. Xin and G. Rouskas, "Multicast routing under optical layer constraints," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Hong Kong, March 2004.

[204] F. Yan, W. Hu, W. Sun, W. Guo, Y. Jin, and H. He, "Allocation of wavelength selective and convertible cross connects in optical multicast networks," in *Proc. Asia Communications and Photonics Conference (ACP)*, Shanghai, China, Nov. 2009.

[205] F. Yan, W. Hu, W. Sun, and H. He, "Wavelength assignment algorithm in optical multicast networks with multi-wavelength conversion," in *Proc. 15th Asia-Pacific Conference on Communications (APCC)*, Shanghai, China, Oct. 2009.

[206] S. Yan, M. Ali, and J. Deogun, "Route optimization of multicast sessions in sparse light-splitting optical networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, San Antonio, TX, Nov. 2001.

[207] D.-N. Yang and W. Liao, "Design of light-tree based logical topologies for multicast streams in wavelength routed optical networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, San Francisco, CA, March 2003.

[208] X. Yang and B. Ramamurthy, "Dynamic routing in translucent WDM optical networks: The intradomain case," *IEEE/OSA Journal of Lightwave Technology*, vol. 23, no. 3, pp. 955 – 971, March 2005.

[209] X. Yang, L. Shen, and B. Ramamurthy, "Survivable lightpath provisioning in WDM mesh networks under shared path protection and signal quality constraints," *IEEE/OSA Journal of Lightwave Technology*, vol. 23, no. 4, pp. 1556 – 1567, Apr. 2005.

[210] K. Yla-Jarkko, M. Leppihalme, S. Tammela, T. Niemi, and A. Tervonen, "Scalability of a metropolitan bidirectional multifiber WDM-ring network," *Springer Photonic Network Communications*, vol. 3, pp. 349–362, October 2001.

[211] K.-M. Yong, Y.-H. Cheng, and G.-S. Poo, "Dynamic multicast routing and wavelength assignment with minimal conversions in delay-constrained WDM networks," in *Proc. 18th Internatonal Conference on Computer Communications and Networks (ICCCN)*, Virgin Islands, Aug. 2009.

[212] K.-M. Yong, G.-S. Poo, and T.-H. Cheng, "Optimal placement of multicast and wavelength converting nodes in multicast optical virtual private network," in *Proc. IEEE Conference on Local Computer Networks (LCN)*, Sydney, Australia, Nov. 2005.

[213] K.-M. Yong, T.-H. Cheng, G. Xiao, and L. Zhou, "Placement of multicast capable nodes in power constrained all-optical WDM networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Miami, FL, Dec. 2010.

[214] O. Yu and Y. Cao, "Dynamic groupcast traffic grooming in WDM networks," in *Proc. IEEE International Conference on Communications (ICC)*, Istanbul, Turkey, June 2006.

[215] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Optical Network Magazine*, vol. 1, pp. 47–60, Jan. 2000.

[216] Y. Zhai, Y. Pointurier, S. Subramaniam, and M. Brandt-Pearce, "Performance of dedicated path protection in transmission-impaired DWDM networks," in *Proc. IEEE International Conference on Communications (ICC)*, Glasgow, Scotland, June 2007.

[217] ——, "QoS-aware RWA algorithms for path-protected DWDM networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2007.

[218] F. Zhang and W. Zhong, "Performance evaluation of p-cycle based protection methods for provisioning of dynamic multicast sessions in mesh WDM networks," *Springer Photonic Network Communications*, vol. 16, no. 2, pp. 127–138, 2008.

[219] F. Zhang and W.-D. Zhong, "Applying p-cycles in dynamic provisioning of survivable multicast sessions in optical WDM networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2007.

[220] H. Zhang, *WDM Mesh Networks: Management and Survivability*. Norwell, MA: Kluwer Academic Publishers, 2003.

[221] S. Zhang, D. Shen, and C.-K. Chan, "Energy efficient time-aware traffic grooming in wavelength routing networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Miami, FL, Dec. 2010.

[222] ——, "Energy-efficient traffic grooming in WDM networks with scheduled time traffic," *IEEE/OSA Journal of Lightwave Technology*, vol. 29, no. 17, pp. 2577 –2584, Sep. 2011.

[223] X. Zhang and C. Qiao, "Wavelength assignment for dynamic traffic in multi-fiber WDM networks," in *Proc. 7th International Conference on Computer Communications and Networks (ICCCN)*, Lafayette, LA, Oct. 1998.

[224] X. Zhang, J. Wei, and C. Qiao, "Constrained multicast routing in WDM networks with sparse light splitting," *IEEE/OSA Journal of Lightwave Technology*, vol. 18, no. 12, pp. 1917 –1927, Dec. 2000.

[225] Z. Zhang and A. Acampora, "A heuristic wavelength assignment algorithm for multihop WDM networks with wavelength routing and wavelength reuse," in *Proc. 13th IEEE International Conference on Computer Communications (INFOCOM)*, Toronto, Canada, June 1994.

[226] Y. Zhao, J. Zhang, H. Zhang, W. Gu, and Y. Ji, "Novel path computation element-based traffic grooming strategy in Internet Protocol over wavelength division multiplexing networks," *IET Communications*, vol. 5, no. 8, pp. 1138 –1146, May 2011.

[227] W.-D. Zhong, F. Zhang, and Y. Jin, "Optimized designs of p-cycles for survivable multicast sessions in optical WDM networks," in *Proc. Second International Conference on Communications and Networking in China (CHINACOM)*, Shanghai, China, Aug. 2007.

[228] F. Zhou, M. Molnar, and B. Cousin, "Avoidance of multicast incapable branching nodes for multicast routing in WDM networks," in *Proc. 33rd IEEE Conference on Local Computer Networks (LCN)*, Montreal, Canada, Oct. 2008.

[229] F. Zhou, M. Molnar, B. Cousin, and G. Simon, "Power optimal design of multicast light-trees in WDM networks," *IEEE Communications Letters*, vol. 15, no. 11, pp. 1240–1242, Nov. 2011.

[230] Y. Zhou and G.-S. Poo, "Multicast wavelength assignment for sparse wavelength conversion in WDM networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Barcelona, Spain, Apr. 2006.

[231] ——, "Multi-wavelength multicast wavelength assignment algorithm for limited wavelength conversion in wavelength-division multiplexing networks," *IET Communications*, vol. 1, no. 4, pp. 776–783, Aug. 2007.

[232] H. Zhu, H. Zang, K. Zhu, and B. Mukherjee, "A novel generic graph model for traffic grooming in heterogeneous WDM mesh networks," *IEEE/ACM Transactions on Networking*, vol. 11, no. 2, pp. 285–299, April 2003.

[233] K. Zhu and B. Mukherjee, "On-line approaches for provisioning connections of different bandwidth granularities in WDM mesh networks," in *Proc. IEEE/OSA Optical Fiber Communication Conference (OFC)*, Anaheim, CA, March 2002.

[234] ——, "Traffic grooming in an optical WDM mesh network," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 1, pp. 122–133, January 2002.

[235] ——, "A review of traffic grooming in WDM optical networks: Architectures and challenges," *SPIE Optical Networks Magazine*, vol. 4, no. 2, pp. 55–64, March/April 2003.

[236] Z.-J. Zhu, W. Dong, and Z.-C. Le, "A novel segment protection with segment route scheme in multicasting survivable networks," in *Proc. Asia Communications and Photonics Conference (ACP)*, Shanghai, China, Nov. 2009.

[237] S. Zsigmond, G. Németh, and T. Cinkler, "Mutual impact of physical impairments and grooming in multilayer networks," in *Proc. International Conference on Optical Networking Design and Modeling (ONDM)*, Athens, Greece, May 2007.

[238] N. Zulkifli, C. Okonkwo, and K. Guild, "Dispersion optimized impairment constraint based routing and wavelength assignment algorithms for all-optical networks," in *Proc. International Conference on Transparent Optical Networks (ICTON)*, Nottingham, UK, June 2006.