

2012

# Optimization of WDM Optical Networks

Quazi R. Rahman  
*University of Windsor*

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>

---

## Recommended Citation

Rahman, Quazi R., "Optimization of WDM Optical Networks" (2012). *Electronic Theses and Dissertations*. 5413.  
<https://scholar.uwindsor.ca/etd/5413>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email ([scholarship@uwindsor.ca](mailto:scholarship@uwindsor.ca)) or by telephone at 519-253-3000ext. 3208.

# OPTIMIZATION OF WDM OPTICAL NETWORKS

By:  
Quazi R Rahman

A Dissertation  
Submitted to the Faculty of Graduate Studies  
Through the School of Computer Science  
in Partial Fulfillment of the Requirements for  
The Degree of Doctor of Philosophy at the  
University of Windsor  
Windsor, Ontario, Canada  
2012

Copyright ©2012 Quazi R Rahman

=====

Optimization of WDM Optical Networks

by  
Quazi R Rahman

APPROVED BY:

---

Dr. Amiya Nayak, External Examiner  
University of Ottawa

---

Dr. Richard Caron, External Reader  
Department of Mathematics and Statistics

---

Dr. Jianguo Lu, Internal Reader  
School of Computer Science

---

Dr. Arunita Jeakel, Internal Reader  
School of Computer Science

---

Dr. Yash Aneja, Advisor  
School of Business

---

Dr. Subir Bandyopadhyay, Advisor  
School of Computer Science

---

Dr. Joseph Casey, Chair of Defence  
Department of Psychology

14 September 2012

# Declaration of Co-Authorship / Previous Publication

## I Co-Authorship Declaration

Chapter 5, Section 5.2 of this dissertation incorporates the outcome of a joint research undertaken in collaboration with Mr. Sujogya Banerjee and Mr. Sudheendra Murthy under the supervision of professor Dr. Arunabha Sen of Arizona State University, USA. For this part of the research, the key ideas, algorithm designs, justifications of methodology and outline of experiments, were performed by the author, and the contribution of coauthors was primarily through the provision of experimental design, carry out the experiments, data analysis and interpretation.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my thesis, and have obtained written permission from each of the co-authors to include the above material in my thesis.

I certify that, with the above qualification, this thesis, and the research to which it refers, is the product of my own work.

## II Declaration of Previous Publication

This thesis includes 4 original papers that have been previously published in peer reviewed conferences as follows:

Thesis Chapter	Publication title/full citation	Publication status
Chapter 3	A Branch, Price and Cut Approach for Optimal Traffic Grooming in WDM Optical Networks; Rahman, Q.; Bandyopadhyay, S.; Aneja, Y.; IEEE International Conf. on Communications, June 2011, DOI: 10.1109/icc.2011.5962606. Copyright ©2011, IEEE. Reprinted with permission from IEEE.	Published
Chapter 5 Section 5.2	Dynamic Lightpath Allocation in Translucent WDM Optical Networks; Bandyopadhyay, S.; Rahman, Q.; Banerjee, S.; Murthy, S.; Sen, A.; IEEE International Conf. on Communications, June 2009, DOI: 10.1109/ICC.2009.5198956. Copyright ©2009, IEEE. Reprinted with permission from IEEE.	Published
Chapter 5 Section 5.3	Path protection in Translucent WDM Optical Networks; Rahman, Q.; Bandyopadhyay, S.; Bari, A.; Jaekel, A.; Aneja, Y.; Distributed Computing and Networking, (LNCS), vol. 5935, 2010, DOI: 10.1007/978-3-642-11322-2_38. Copyright ©2010, Springer Berlin/Heidelberg. Reprinted with permission from Springer.	Published
Chapter 5 Section 5.4	On Static RWA in Translucent Optical Networks Rahman, Q.; Bandyopadhyay, S.; Aneja, Y.; IEEE Symposium on Computers and Comm., July 2012, DOI: 10.1109/ISCC.2012.6249288. Copyright ©2012, IEEE. Reprinted with permission from IEEE.	Published

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as graduate student at the University of Windsor.

I declare that, to the best of my knowledge, my thesis does not infringe upon anyones copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my the-

sis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

# Abstract

Optical network, with its enormous data carrying capability, has become the obvious choice for today's high speed communication networks. Wavelength Division Multiplexing (WDM) technology and Traffic Grooming techniques enable us to efficiently exploit the huge bandwidth capacity of optical fibers. Wide area translucent networks use sparse placement of regenerators to overcome the physical impairments and wavelength constraints introduced by all optical (transparent) networks, and achieve a performance level close to fully switched (opaque) networks at a much lesser network cost.

In this dissertation we discuss our research on several issues on the optimal design of optical networks, including optimal traffic grooming in WDM optical networks, optimal regenerator placement problem (RRP) in translucent networks, dynamic lightpath allocation and dynamic survivable lightpath allocation in translucent networks and static lightpath allocation in translucent networks. With extensive simulation experiments, we have established the effectiveness and efficiencies of our proposed algorithms.

# Dedication

*To my:*

*Mother: Hurun Nahar*

*Father: Quazi Talebur Rahman*

*Wife: Afshana Tajmin Reshad*

*– I am so blessed to have them in my life.*



# Acknowledgements

It gives me immense pleasure to thank a few very important people who have helped me a lot during the whole process of my graduate studies.

At first, I would like to express my sincere gratitude to my supervisors, Dr. Subir Bandyopadhyay and Dr. Yash Aneja, for their constant guidance and extensive support throughout my graduate studies. This work could not have been achieved without their continuous encouragement, astute advices, suggestions and cooperation.

I would like to thank Dr. Amiya Nayak for his kind acceptance to be the examiner of my thesis defence.

I would like to thank members of my Ph.D. thesis committee, Dr. Richard Caron, Dr. Arunita Jaekel and Dr. Jianguo Lu for their instructive advices, suggestions and comments.

I would also like to thank all the faculty and staff members at the School of Computer Science, for their cordial supports and helps, as well as everybody else who has offered any help, during my graduate study at the University of Windsor.

Finally, I would like to thank my wife, Afshana Reshad, for her endless love and care and for always being there for me.

# Contents

Declaration of Co-Authorship / Previous Publication . . . . .	iii
Abstract . . . . .	vi
Dedication . . . . .	vii
Acknowledgements . . . . .	viii
List of Tables . . . . .	xiv
List of Figures . . . . .	xvii
List of Symbols . . . . .	xviii
List of Acronyms . . . . .	xxix
<b>1 Introduction</b>	<b>1</b>
1.1 Optical Networks . . . . .	1
1.2 Impairments in Optical Networks . . . . .	3
1.2.1 Translucent Networks . . . . .	3
1.3 Motivation . . . . .	4
1.4 Problems Addressed in This Dissertation . . . . .	5
1.4.1 Optimal Traffic Grooming in WDM Networks . . . . .	5
1.4.2 Optimal Regenerator Placement in Translucent Networks	7
1.4.3 Optimal Lightpath Allocation in Translucent Networks	7
1.5 Solution Outline and Contributions . . . . .	9

1.5.1	A Branch, Price and Cut Approach for Optimal Traffic Grooming . . . . .	9
1.5.2	A Branch and Cut Approach for Optimal Regenerator Placement . . . . .	11
1.5.3	Optimal Lightpath Allocation . . . . .	11
1.5.3.1	Dynamic Lightpath Allocation . . . . .	12
1.5.3.2	Dynamic Survivable Lightpath Allocation . . . . .	12
1.5.3.3	Static Lightpath Allocation . . . . .	12
1.6	Thesis Organization . . . . .	13
<b>2</b>	<b>Review on Related Topics</b>	<b>14</b>
2.1	WDM Optical Networks . . . . .	14
2.1.1	Optical Fibers . . . . .	15
2.1.2	WDM Technology . . . . .	16
2.1.3	Optical Add-Drop Multiplexers (OADM) . . . . .	18
2.1.4	Wavelength Routers ( $\lambda$ Routers) . . . . .	19
2.1.5	Optical Transmitters and Receivers . . . . .	20
2.1.6	Wavelength Converters . . . . .	20
2.1.7	Physical Topology . . . . .	21
2.1.8	Logical Topology . . . . .	22
2.1.9	Traffic Grooming . . . . .	23
2.1.9.1	Traffic Matrix . . . . .	26
2.1.10	Routing and Wavelength Assignment (RWA) . . . . .	27
2.1.11	Fault Management in WDM Optical Networks . . . . .	28
2.1.11.1	Fault Management Techniques . . . . .	29
2.1.12	Physical Layer Impairments in Optical Networks . . . . .	30

2.1.13	Wide Area Optical Networks . . . . .	31
2.1.13.1	Opaque Networks . . . . .	33
2.1.13.2	Transparent Networks . . . . .	33
2.1.14	Translucent Networks . . . . .	34
2.1.14.1	Translucent Lightpath . . . . .	34
2.1.15	Designing Translucent Networks . . . . .	35
2.1.15.1	Regenerator Placement Problem . . . . .	37
2.1.15.2	Routing with Regenerators Problem . . . . .	40
2.2	Some Useful Operations Research (OR) Techniques . . . . .	41
2.2.1	Branch and Bound Algorithm . . . . .	42
2.2.2	Branch and Price . . . . .	44
2.2.3	Branch and Cut . . . . .	44
2.2.4	Branch, Price and Cut . . . . .	46
2.2.5	Knapsack Problem . . . . .	46
2.2.5.1	0-1 Knapsack Inequalities . . . . .	47
2.2.6	Arc-chain Representation . . . . .	48
<b>3</b>	<b>Optimal Traffic Grooming in WDM Networks</b>	<b>51</b>
3.1	Introduction . . . . .	51
3.2	An ILP for Non-bifurcated Traffic Grooming . . . . .	53
3.2.1	Justification for the ILP Formulation . . . . .	54
3.3	Traffic Grooming using Arc-chain Formulation . . . . .	55
3.4	Traffic Grooming using Branch, Price and Cut . . . . .	56
3.4.1	An LP for Bifurcated Traffic Grooming . . . . .	59
3.4.1.1	Justification for the LP Formulation . . . . .	61
3.4.2	Advantages of the Arc-Chain Representation . . . . .	61

3.4.3	The Pricing Policy . . . . .	63
3.4.4	The Branching Strategy . . . . .	68
3.5	Experimental Results . . . . .	70
<b>4</b>	<b>Optimal Regenerator Placement in Translucent Networks</b>	<b>75</b>
4.1	Introduction . . . . .	75
4.2	A Compact Formulation to Solve the RPP . . . . .	78
4.2.1	Justification for the Compact Formulation . . . . .	79
4.3	RPP using Branch and Cut Algorithm . . . . .	80
4.3.1	An ILP Formulation to Solve the RPP . . . . .	81
4.3.2	A branch and Cut Scheme to Solve the RPP . . . . .	86
4.4	Experimental Results . . . . .	92
<b>5</b>	<b>Optimal Lightpath Allocation in Translucent Networks</b>	<b>97</b>
5.1	Introduction . . . . .	97
5.2	RRP for Dynamic Lightpath Allocation . . . . .	100
5.2.1	DLA: An ILP Formulation for Optimal Solution . . . . .	100
5.2.1.1	Justification for the Formulation DLA . . . . .	103
5.2.2	A Heuristic for Dynamic Lightpath Allocation . . . . .	104
5.2.3	Experimental Results . . . . .	110
5.3	RRP for Dynamic Survivable Lightpath Allocation . . . . .	115
5.3.1	DSLAs-1: An ILP Formulation for Optimal Solution . . . . .	117
5.3.1.1	Justification for the Formulation DSLAs-1 . . . . .	123
5.3.2	DSLAs-2: A Fast ILP Formulation . . . . .	126
5.3.2.1	Justification for the Formulation DSLAs-2 . . . . .	132
5.3.3	Experimental Results . . . . .	134
5.4	Static Lightpath Allocation in Translucent Networks . . . . .	138

5.4.1	SLA: An ILP Formulation for Optimal Solution . . . . .	141
5.4.1.1	Justification for the Formulation SLA . . . . .	144
5.4.2	A Heuristic for Static Lightpath Allocation . . . . .	146
5.4.3	ILPH: An ILP Formulation used in the Heuristic . . . . .	149
5.4.3.1	Justification for the Formulation ILPH . . . . .	151
5.4.4	Experimental Results . . . . .	152
<b>6</b>	<b>Conclusions and Future Works</b>	<b>155</b>
6.1	Conclusions . . . . .	155
6.2	Future Works . . . . .	156
	<b>Appendix A: Copyright Releases</b>	<b>158</b>
	<b>Bibliography</b>	<b>166</b>
	<b>Vita Auctoris</b>	<b>180</b>

# List of Tables

2.1	An example of a traffic matrix . . . . .	27
4.1	Comparing average execution time using FRM-1 and FRM-2 . . . . .	94
4.2	Average execution time for large networks using FRM-2 . . . . .	95
5.1	Comparison of SLA and Heuristic Performances . . . . .	153
5.2	Heuristics Results for Large Networks . . . . .	153

# List of Figures

2.1	Cross-section of an optical fiber . . . . .	16
2.2	Propagation of optical signal using total internal reflection . .	16
2.3	Signal bandwidth and channel spacing (modified from [58]). . .	18
2.4	Schematic diagram of an OADM . . . . .	18
2.5	Wavelength Router (Optical Cross Connect) . . . . .	19
2.6	Optical Transceivers (modified from [77]). . . . .	20
2.7	A 4-node network and corresponding physical topology. . . . .	22
2.8	Logical topology of a 4-node network. . . . .	22
2.9	An opaque optical network . . . . .	33
2.10	A transparent optical network . . . . .	34
2.11	A translucent optical network . . . . .	35
2.12	A translucent lightpath . . . . .	36
2.13	Long haul optical network with distances between the nodes in km. . . . .	37
2.14	A network with four nodes . . . . .	49
3.1	An example showing how branching can be used for the traffic grooming problem . . . . .	69



3.2	Comparison of execution time required by BPC and CPLEX on different size requests. . . . .	72
3.3	Comparison of execution time required by BPC and CPLEX on different size requests. . . . .	72
3.4	Comparison of execution time required by BPC and CPLEX on different size requests. . . . .	73
3.5	Percentage of execution time required by BPC over CPLEX for different size of requests and for different networks. . . . .	73
4.1	Reachability graph for a 6 node network . . . . .	81
4.2	Extended reachability graph created from reachability graph . . . . .	84
4.3	Comparing average execution time for medium density networks . . . . .	94
4.4	Illustrating FRM-2 execution time for large networks . . . . .	95
5.1	A simple translucent network with fiber lengths in km . . . . .	99
5.2	Effect of varying the number of pre-computed paths considered . . . . .	111
5.3	Traffic load vs. call blocking probability for LATA ‘X’ . . . . .	112
5.4	Traffic load vs. call blocking probability for ARPANET . . . . .	112
5.5	Number of regenerator vs. call blocking probability for ARPANET . . . . .	113
5.6	Optical reach vs. blocking probability for ARPANET . . . . .	114
5.7	Average number of regenerators computed by the heuristic and DLA for different networks . . . . .	115
5.8	Blocking probabilities using DSLA-1 and DSLA-2 for 40-node networks. . . . .	135
5.9	Individual lightpath deployment time using DSLA-1 and DSLA-2 for 40-node networks. . . . .	135

5.10	Number of regenerators required by DSLA-1 and DSLA-2 for 24-node networks. . . . .	136
5.11	Average lightpath deployment time using DSLA-1 and DSLA-2 for different networks. . . . .	136
5.12	Average blocking probability using DSLA-1 and DSLA-2 for different networks. . . . .	137
5.13	A wide area translucent network with distances between the nodes in km. . . . .	139
5.14	Percentage of variances of outputs using Heuristic and SLA . .	154

# List of Symbols

$a_{ij}^l$  : a constant defined as follows

$$a_{ij}^l = \begin{cases} 1 & \text{if the } l^{\text{th}} \text{ existing primary lightpath} \\ & \text{uses the physical edge } i \rightarrow j, \\ 0 & \text{otherwise.} \end{cases}$$

$A_p$  : a binary variable defined as follows:

$$A_p = \begin{cases} 1 & \text{if the source-destination pair } p \text{ is selected to} \\ & \text{handle the new request for communication,} \\ 0 & \text{otherwise.} \end{cases}$$

$\mathcal{A}$  : a constraints matrix.

$\overrightarrow{\mathcal{A}}^p$  : column  $p$  in the constraints matrix  $\mathcal{A}$ .

$AC$  : an arc-chain incidence matrix.

$b_{ij}^{lh}$  : a constant defined as follows

$$b_{ij}^{lh} = \begin{cases} 1 & \text{if segment } h \text{ of the } l^{\text{th}} \text{ existing backup lightpath} \\ & \text{uses the physical edge } i \rightarrow j, \\ 0 & \text{otherwise.} \end{cases}$$

$B$  : a basis for the revised simplex method.

$B_{init}$  : basis for the initial feasible solution.

$B_{opt}$  : basis for the optimal solution.

$B_{opt}^i$  : basis for the  $i^{th}$  branch of the optimal solution.

$B_{scs}$  : basis for the single column solution where each commodity  $k, i \leq k \leq K$  is using only one chain.

$C$  : a set called cover.

$C'$  : a proper subset of cover  $C$ .

$\mathcal{C}$  : a set of paths from the current node.

$\vec{C}_j^k$  : the  $j^{th}$  chain of the  $k^{th}$  commodity — a vector of 1's and 0's.

$\vec{c}_B$  : a vector of  $(H + m + K)$  cost coefficients, each coefficient corresponding to a variable in the basis.

$d_{ij}$  : a constant denoting the length of the physical edge  $i \rightarrow j$ .

$\mathbb{D}$  : a Disconnecting Set of Nodes (DSN).

$D$  : the destination node of a request for which a new lightpath has to be setup.

$D^k$  : the destination node for commodity  $k$ .

$e_h$  : the edge associated with the  $h^{th}$  lifting constraint,  $e_h, 1 \leq h \leq H$ .

$E$  : the set of all node pairs  $(i, j)$  such that there is a directed edge  $i \rightarrow j$  from node  $i$  to node  $j$  in the physical topology, representing a fiber from node  $i$  to node  $j$ .

$E_L$  : the set of all node pairs  $(i, j)$  such that there is a logical edge  $i \rightarrow j$  from node  $i$  to node  $j$  in the logical topology, representing a lightpath from node  $i$  to node  $j$ .

$E_R$  : set of all edges in the reachability graph.

$E_s$  : set of physical edges, that constitutes a segment of a translucent lightpath, with a total length  $\leq r$ .

$E_R^X$  : set of all edges in the extended reachability graph.

$\mathbb{E}_P$  : set of all edges in the path intersection graph.

$F_{ij}^k$  : a binary flow variable for each commodity  $k \in K$  and each logical edge  $(i, j) \in E_L$ , such that

$$F_{ij}^k = \begin{cases} 1 & \text{if the commodity } k \text{ uses logical edge } (i, j), \\ 0 & \text{otherwise.} \end{cases}$$

$f_{ij}^k$  : a continuous flow variable indicating the amount of flow for commodity  $k \in K$  over the edge  $(i, j) \in E_R$  in the reachability graph.

$\varphi_{ij}^k$  : a binary flow variable for each commodity  $k \in K$  and for each physical edge  $(i, j) \in E$ , such that

$$\varphi_{ij}^k = \begin{cases} 1 & \text{if commodity } k \text{ uses edge } (i, j), \\ 0 & \text{otherwise.} \end{cases}$$

$G$  : the physical topology represented by a graph such that  $G = \{N, E\}$

$G_L$  : the logical topology represented by a graph such that  $G_L = \{N, E_L\}$

$G_L^k$  : a directed graph for commodity  $k$  such that  $G_L^k = \{N, E_L\}$

$G_R$  : the physical topology represented by the reachability graph such that  $G_R = \{N, E_R\}$

$G_R^X$  : the physical topology represented by the extended reachability graph  
such that  $G_R^X = \{V_R^X, E_R^X\}$

$G_P$  : the path intersection graph such that  $G_P = \{V_P, E_P\}$

$H$  : the number of lifting constraints included in the basis.

$K$  : the set of all commodities representing all source-destination pairs  $(S^k, D^k)$   
being considered in the formulation.

$K^i$  : the  $i^{th}$  commodity.

$\ell_h^{jk}$  : entry for commodity  $k$  in  $h^{th}$  lifting constraint defined as:

$$\ell_h^{jk} = \begin{cases} \xi_h^k & \text{if edge } e_h \text{ appears in chain } j \text{ of commodity } k, \\ 0 & \text{otherwise.} \end{cases}$$

$L$  : lower bound for an optimal solution.

$L^i$  : lower bound of the optimal solution for the  $i^{th}$  subproblem.

$\mathcal{L}$  : total number of logical edges already established.

$\mathfrak{L}_h$  : right side of the  $h^{th}$  lifting constraint.

$m$  : number of logical edges in the logical topology.

$\bar{m}$  : the number of paths, for transparent lightpaths, that will be pre-computed  
between every node pairs.

$M$  : a large positive number.

$n^k$  : total number of chains for commodity  $k$ .

$\hat{n}$  : the total number of chains for all commodities.

$\mathcal{N}$  : the set of all end-nodes in the physical network.

$\mathcal{N}_S$  : a set of new states to explore.

$\mathbf{p}$  : a transparent path.

$\mathbf{P}$  : a problem formulation.

$\mathcal{P}^k$  : a transparent lightpath for the commodity  $k$ .

$\mathbb{P}$  : the set of paths, each with a total length  $\leq r$ , that may be used to reach  $x$  from source  $S$ .

$\mathbb{P}_{xy}$  : the set of pre-computed paths from node  $x$  to  $y$ .

$\mathcal{Q}$  : set of all wavelength channels available in a fiber.

$\overline{\mathcal{Q}}_{ij}$  : the set of channels still available on edge  $i \rightarrow j$ .

$r$  : the optical reach of a network.

$\mathbf{R}$  : a set of nodes.

$\mathbb{R}$  : the set of all pre-computed source-destination pairs  $(i, j)$  in the physical topology, each of length  $\leq r$ , such that a transparent lightpath may be set up from  $i$  to  $j$ .  $\mathbb{R}$  includes  $\overline{m}$  edge-disjoint routes from  $i$  to  $j$ .

$s$  : a constant denoting the maximum possible number of segments in a translucent lightpath.

$s(\mathbf{p})$  : the starting node of the PTSP  $\mathbf{p}$ .

$s^{relaxed}$  : a relaxed LP solution

$\mathbf{S}$  : the starting node of a request from which a new lightpath has to be setup.

$\mathcal{S}^k$  : the source node for commodity  $k$ .

$\mathcal{S}$  : a superset of all feasible solutions.

$\mathbb{S}$  : a set of states that need to be explored.

$t(p)$  : the terminating node of the PTSP  $p$ .

$u_j^k$  : a variable denoting the fractional flow of commodity  $k$  in the  $j^{\text{th}}$  chain,  
 $0 \leq u_j^k \leq 1$ .

$u^k$  : the flow of commodity  $k$  on the saturated edge  $e_h$  for which the  $h^{\text{th}}$  lifting constraint has been determined.

$U$  : upper bound for an optimal solution.

$U^i$  : upper bound for the  $i^{\text{th}}$  branch of the optimal solution.

$\mathbb{U}$  : edges in the min-cut obtained from max-flow algorithm.

$v$  : a vertex of a path intersection graph.

$V_R^X$  : set of vertexes in the extended reachability graph.

$\mathbb{V}_P$  : set of vertexes in the path intersection graph.

$w_{ij}^q$  : a constant defined as follows

$$w_{ij}^q = \begin{cases} 1 & \text{if an existing lightpath uses channel } q \text{ on edge } i \rightarrow j, \\ 0 & \text{otherwise.} \end{cases}$$

$W_q^k$  : a binary variable for the new lightpath (primary lightpath) for dynamic (dynamic survivable) RWA, defined as follows:

$$W_q^k = \begin{cases} 1 & \text{if the } k^{\text{th}} \text{ segment of the new lightpath} \\ & \text{(primary lightpath) uses channel } q, \\ 0 & \text{otherwise.} \end{cases}$$



$\mathcal{W}_v$  : the set of colors to color vertex  $v$  of  $\mathbb{G}_P$ .

$\mathcal{W}$  : the set of the set of colors  $\{\mathcal{W}_v : v \in V_P\}$ .

$\mathbb{W}_p^{gq}$  : a binary variable defined as follows

$$\mathbb{W}_p^{gq} = \begin{cases} 1 & \text{if the new primary lightpath uses channel } q \text{ in the} \\ & g^{th} \text{ route of the source-destination pair } p, \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbf{x}$  : a node in the optical network.

$\mathbf{X}_{ij}^k$  : a binary variable for the new lightpath (primary lightpath) for dynamic (dynamic survivable) RWA, defined as follows:

$$X_{ij}^k = \begin{cases} 1 & \text{if the } k^{th} \text{ segment of the new lightpath} \\ & \text{(primary lightpath) uses edge } i \rightarrow j \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbb{X}_p^g$  : a binary variable defined as follows:

$$\mathbb{X}_p^g = \begin{cases} 1 & \text{if the } g^{th} \text{ pre-computed route is selected to realize the segment} \\ & \text{of the new primary lightpath corresponding to the} \\ & \text{source-destination pair } p \in \mathcal{P}, \\ 0 & \text{otherwise.} \end{cases}$$

$\vec{\mathbf{x}}^k$  : the column vector of variables, containing  $n^k$  elements  $x_1^k, x_2^k, \dots, x_{n^k}^k$ .

$\mathbf{x}_s^i$  : the  $i^{\text{th}}$  slack variable.

$\mathbf{x}_a^i$  : the  $i^{\text{th}}$  artificial variable corresponding to  $i$ -th lifting constraint.

$\vec{\mathbf{x}}_B$  : the vector corresponding to the variables in the basis.

$y^k$ : an integer variable for lightpath  $k \in K$ , denoting the minimum number of segments needed for lightpath  $k$ .

$Y_{ij}^k$  : a binary variable for the new backup lightpath defined as follows:

$$Y_{ij}^k = \begin{cases} 1 & \text{if the } k^{th} \text{ segment of the new backup} \\ & \text{lightpath uses the edge } i \rightarrow j \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbb{Y}_p^g$  : a binary variable defined as follows:

$$\mathbb{Y}_p^g = \begin{cases} 1 & \text{if the } g^{th} \text{ pre-computed route is selected to realize the segment} \\ & \text{of the new backup lightpath corresponding to the} \\ & \text{source-destination pair } p \in \mathcal{P}, \\ 0 & \text{otherwise.} \end{cases}$$

$\vec{y}$  : a vector containing the values of simplex multipliers.

$z$  : global objective value of the feasible integer solution.

$z^i$  : objective value of the feasible integer solution of the  $i^{th}$  subproblem.

$z_{ij}^q$  : a constant defined as follows

$$z_{ij}^q = \begin{cases} 1 & \text{if an existing backup lightpath uses} \\ & \text{channel } q \text{ on edge } i \rightarrow j, \\ 0 & \text{otherwise.} \end{cases}$$

$Z_q^k$  : a binary variable for the new backup lightpath, defined as follows:

$$Z_q^k = \begin{cases} 1 & \text{if the } k^{th} \text{ segment of the new backup} \\ & \text{lightpath uses channel } q, \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbb{Z}_p^{gq}$  : a binary variable for the new backup lightpath, defined as follows

$$\mathbb{Z}_p^{gq} = \begin{cases} 1 & \text{if the new backup lightpath uses channel } q \text{ in the } g^{th} \text{ route} \\ & \text{of the source-destination pair } p, \\ 0 & \text{otherwise.} \end{cases}$$

$\alpha_{ij}^{pg}$  : a constant defined as follows:

$$\alpha_{ij}^{pg} = \begin{cases} 1 & \text{if the } g^{th} \text{ route of the source-destination pair } p \\ & \text{includes physical edge } i \rightarrow j, \\ 0 & \text{otherwise.} \end{cases}$$

$\alpha_k$  : the simplex multiplier, corresponding to the commodity  $k$ .

$\beta_i$  : a binary variable for each node  $i \in N$  in the physical topology, such that

$$\beta_i = \begin{cases} 1, & \text{if node } i \text{ is made a regenerator node,} \\ 0, & \text{otherwise.} \end{cases}$$

$\beta_i^k$  : a binary variable for each commodity  $k \in K$  and each node  $i \in N$  in the physical topology, such that

$$\beta_i^k = \begin{cases} 1, & \text{if commodity } k \text{ is regenerated at node } i, \\ 0, & \text{otherwise.} \end{cases}$$

$\vec{\beta}^*$  : a vector for the objective variable values from an optimal solution.

$\vec{\beta}^i$  : a vector for the objective variable LP solution values of the  $i^{th}$  subproblem.

$\eta_p^g$  : the set of available channel numbers that can be used to set up a new primary lightpath using the route  $g$  of the source-destination pair  $p$ .

$\gamma_l^k$  : a non-negative continuous variable for the new backup path whose values are restricted by the constrains, such that

$$\gamma_{lh}^k = \begin{cases} 1 & \text{if the new backup path in its segment } k \\ & \text{shares both an edge and a channel number with} \\ & \text{the } h^{th} \text{ segment of } l^{th} \text{ existing backup path,} \\ 0 & \text{otherwise.} \end{cases}$$

$\delta_i$  : a constant for node  $i \in E_N$ , defined as follows:

$$\delta_i = \begin{cases} 1 & \text{if the } i^{\text{th}} \text{ node is a 3R regenerator node,} \\ 0 & \text{otherwise.} \end{cases}$$

$\kappa_{gg}^{ph}$  : a constant defined as follows:

$$\kappa_{gg}^{ph} = \begin{cases} 1 & \text{if the } g^{\text{th}} \text{ route of the source-destination pair } p \text{ and the } h^{\text{th}} \\ & \text{route of the source-destination pair } q \text{ share some edge(s),} \\ 0 & \text{otherwise.} \end{cases}$$

$\lambda_{ph}^{gl}$  : a non-negative continuous variable for the new backup lightpath whose values are restricted by the constrains, such that

$$\lambda_{ph}^{gl} = \begin{cases} 1 & \text{if the } g^{\text{th}} \text{ route for the source-destination pair } p \text{ shares an} \\ & \text{edge with the segment } h \text{ of the } l^{\text{th}} \text{ backup lightpath,} \\ & \text{and the channel number } Z_p^{gq} \text{ matches with } \omega_p^l, \\ 0 & \text{otherwise.} \end{cases}$$

$\Lambda_{max}$  : congestion of the network, defined by the total amount of traffic flowing through the edge carrying the maximum traffic.

$\Psi_{max}$  : congestion of the network, defined by the number of lightpaths flowing through the edge carrying the maximum number of lightpaths.

$\mu_i^{jk}$  : the  $i^{\text{th}}$  element of the  $j^{\text{th}}$  chain of commodity  $k$  so that

$$\mu_i^{jk} = \begin{cases} \Upsilon^k & \text{if edge } i \text{ is in chain } j \text{ of commodity } k, \\ 0 & \text{otherwise.} \end{cases}$$

$\xi_h^k$  : The constraint coefficient of commodity  $k$  in lifting constraint  $h$ .

$\pi_j$  : the simplex multiplier, corresponding to the logical edge  $j$ ,  $1 \leq j \leq m$ .

$\wp(S, D)$  : the shortest path, in the logical topology, from  $S$  to  $D$ .

$\rho_h$  : the simplex multiplier corresponding to lifting constraint  $h$ ,  $1 \leq h \leq H$ .

$\sigma_{ij}^k$  : the weight of edge  $i$  when considering chain  $j$  for commodity  $k$ , defined

as:

$$\sigma_{ij}^k = \begin{cases} 1, & \text{if chain } j \text{ for commodity } k \text{ uses edge } i, \\ 0, & \text{otherwise.} \end{cases}$$

$\Upsilon^k$  : required amount of data communication for commodity  $k \in K$  using  $OC$ - $n$  notation.

$\phi_{hi}^{jk}$  : a weight for edge  $i$  in  $j^{th}$  chain for the commodity  $k$ , corresponding to lifting constraint  $h$  defined as:

$$\phi_{hi}^{jk} = \begin{cases} \xi_h^k & \text{if edge } i = e_h, \\ 0 & \text{otherwise.} \end{cases}$$

$\omega_p^l$  : channel number used by the  $l^{th}$  existing backup lightpath in segment (or source-destination pair)  $p$ .

$\Omega$  : A set of all DSNs.

# List of Acronyms

ASE - Amplified Spontaneous Emission

BAC - Branch And Cut

BER - Bit Error Rate

BILP - Binary Integer Linear Program

BPC - Branch Price and Cut

CD - Chromatic Dispersion

CDS - Connected Dominating Set

CI - Confidence Interval

DLA - Dynamic Lightpath Allocation

DSL A - Dynamic Survivable Lightpath Allocation

DSN - Disconnecting Set of Nodes

DWDM - Dense Wavelength Division Multiplexing

FC - Filter Concatenation

FWM - Four Wave Mixing

LED - Light Emitting Diode

ILP - Integer Linear Program

LCDS - Labeled Connected Dominating Set

LP - Linear Program

LT - Logical Topology

MCDS - Minimum Connected Dominating Set

MCNF - Multi Commodity Network Flow

MPC - Minimum Path Concatenation

OADM - Optical Add Drop Multiplexer

OEO - Optical Electronic Optical

OTN - Optical Transport Network

OXC - Optical Cross Connect

PMD - Polarization Mode Dispersion

PT - Physical Topology

PSTP - Pre-computed Set of Transparent Paths

QoT - Quality of Transmission

RDC - Requests for Data Communication

RPP - Regenerator Placement Problem

RRP - Routing with Regenerator Problem

RWA - Routing and Wavelength Assignment

SAP - Steiner Arborescence Problem

SLA - Static Lightpath Allocation

SONET - Synchronous Optical Networking

SPM - Self Phase Modulation

TWAN - Translucent Wide Area Network

WAN - Wide Area Network

WDM - Wavelength Division Multiplexing

XPM - Cross Phase Modulation



# Chapter 1

## Introduction

### 1.1 Optical Networks

For the past couple of decades we are observing an exponential growth of telecommunication network, which is mostly driven by an ever-increasing user demands for new applications as well as continuous advancements in the technologies involved. With the introduction of optical networks, where *optical fibers*, with their huge bandwidth capacity, serve as the data communication medium, today's telecommunication networks can easily handle the unprecedented bandwidth demand of the modern day communications [3].

The potential data carrying capacity of a single optical fiber is nearly 50 Terabits per second (Tbps) [3]. Introduction of *wavelength division multiplexing* (WDM) technology can efficiently exploit this huge bandwidth capacity of optical fibers. With WDM technology, multiple optical signals can be transmitted, simultaneously and independently, using non-overlapping carrier wavelengths over a single fiber, each at a rate of a few Giga bits per second, which significantly increases the usable bandwidth of an optical fiber [93].

A WDM optical network consists of a set of *end-nodes* (any device that produces or consumes data traffic can be an end-node) each equipped with optical devices, such as *optical transmitters* and *receivers*, *wavelength routers* (also known as *optical cross-connects* (OXC)), *optical add/drop multiplexers* (OADM) etc, and which are interconnected by a set of optical fibers. This configuration defines the *physical topology* [57] of an optical network.

A *logical topology* [57] may be defined over a physical topology by establishing *lightpaths* between the end-nodes [49]. A lightpath is a point-to-point connection, at the optical level, that allows a transmitter at a source node to communicate with a receiver at a destination node using encoded optical signals [62]. A lightpath is allowed to pass through any set of intermediate nodes, as necessary, using optical cross connects (OXCs). If a WDM network has no *wavelength converter* (an optical device that can change the carrier wavelength of an optical signal) at any end-node, a lightpath must use the same carrier wavelength on all links that it traverses [40]. This is known as the *wavelength continuity constraint* [11]. Due to the limitations of the related technologies and the costs involved, most networks today enforce the wavelength continuity constraint. Once the logical topology has been achieved by establishing necessary lightpaths, the physical topology is irrelevant for determining a traffic routing strategy to handle the traffic demands between end-nodes.

The design of a logical topology involves determining the set of lightpaths needed to meet the traffic demands between pairs of end nodes, appropriate routing of the lightpaths over the physical topology (known as the *Routing and Wavelength Assignment* (RWA)) and the proper routing of traffic demands over the logical topology. For a given physical topology of a network and the set of lightpaths to be established, the RWA problem is to select, for

each connection, a suitable path and a carrier wavelength among the many possible choices, so that no two lightpaths sharing a link are assigned the same wavelength [52].

## 1.2 Impairments in Optical Networks

A modern day optical network may have 100's, even 1000's of kilometers long optical fibers connecting individual nodes. (For the rest of this dissertation, we will use the terms "node" and "end-node" interchangeably). Optical signals passing through such long haul or extra long haul fibers can only travel up to a certain distance retaining their characteristics. In a wide-area backbone optical network, spanning a large geographical area, all end-to-end connections, using current technology, cannot be established entirely in the optical domain. Factors, such as *optical noise*, *chromatic dispersion*, *nonlinear effects*, *polarization mode dispersion (PMD)* and *crosstalk* cause the quality of an optical signal to degrade as it propagates through such fibers [70, 73]. An optical network is called a *transparent* network, if all the lightpaths in the network always remain in the optical domain from their respective sources to the corresponding destinations. In an *opaque* network, each lightpath must undergo optical-electronic-optical conversion in every intermediate node it passes through. The notion of *translucent* networks, introduced by Ramamurthy *et al.* in [86, 87], have features of both transparent and opaque networks.

### 1.2.1 Translucent Networks

The distance an optical signal can propagate, before its quality degrades below a threshold level that necessitates the restoration of the signal, is called the

*optical reach*, which typically ranges from 800 to 3000 kilometers [13]. To establish any communication path beyond the optical reach, it is necessary to *reamplify, reshape* and *retime* the optical signal, which is often called the *3R-regeneration* [27, 70]. In a *translucent network*, the optical signal is *regenerated* at the regeneration points (typically a selected subset of the network nodes are capable of signal regeneration), so that the signal may be communicated over long distances. A lightpath established for communication between a source node to a destination node that involves one or more stages of regeneration, are often called a *translucent lightpath*.

### 1.3 Motivation

When designing optimal WDM optical networks, one popular approach is to use *multi-commodity network flow programming* (MCNF) [1] using *Linear Programs* (LP) or *Mixed Integer Linear Programs* (MILP). Such LP or MILP formulations are typically solved using some commercially available mathematical optimization tool, such as the CPLEX [36]. It is well-known [82] that, in general, the number of integer variables in a MILP is crucially important, since the time needed to solve the formulation increases, in general, exponentially with the increase of the number of integer variables. As a consequence, most MILP's for designing optical networks work only for small networks and designers have to use heuristics for larger networks where the margin of error is typically unknown. In many cases, these heuristic approaches are validated only through simulation experiments.

Within the Operations Research (OR) community, integer programming, using cutting plane and branch and bound techniques, have been widely in-

investigated [7, 44]. Many interesting and practical problems possess special structures and exploiting such special structures gives the greatest chance of success with large problems [7]. It has been observed that, by tailoring the algorithm to the special structures of such problems, it may be possible to solve problems involving a very large number of integer variables [82].

We have identified many important problems in WDM network design (e.g., regenerator placement in translucent networks, route and wavelength Assignment (RWA) in transparent as well as translucent networks, traffic grooming in transparent networks) where the techniques for solving Integer Linear programs, such as, *Branch and Cut* or *Branch Price and Cut* [82] may be used to take advantage of the special properties of these problems. With appropriate adaptations, the use of the techniques leads to significant improvements in the time required to find an optimum solution, compared to traditional mathematical optimizations using MCNF programming.

## 1.4 Problems Addressed in This Dissertation

The objective of this dissertation is to present new and efficient algorithms and heuristics to optimally solve the following design problems in WDM optical networks.

### 1.4.1 Optimal Traffic Grooming in WDM Networks

A single lightpath [3, 58] typically supports data rates between 2.5 to 10 gigabits per second, depending on the technology used. Individual requests for data communication are at much lower rates, typically a few megabits per second. Therefore, to ensure effective resource utilization, it is essential to share

the capacity of a lightpath among several low-speed requests. *Traffic Grooming* [24, 34, 37, 38, 97–100] in WDM optical networks is defined as a family of techniques for combining a number of low-speed traffic streams from users, so that the high capacity lightpaths may be used as efficiently as possible. *Static traffic grooming* [97] is used when the traffic requests are known in advance and do not change significantly over relatively long periods of time. In this case, it is reasonable to spend a considerable time to determine an optimal grooming strategy.

There are two models for traffic grooming - the *non-bifurcated* and the *bifurcated* model [3]. In the non-bifurcated (bifurcated) model, the data stream corresponding to any given request is communicated using a single (one or more) logical path(s) from the source of the data stream to its destination. If a bifurcated model is used, routing on a logical topology requires only an LP formulation which can be easily handled [3] using any commercial mathematical programming packages such as the CPLEX. If the non-bifurcated model is used, the additional requirement that each request has to be routed on the logical topology using exactly one logical path means that a formulation for routing over a logical topology has to involve binary variables. In other words, solving an ILP formulation for the non-bifurcation model is more complex and time-consuming. However, as pointed out in [81], bifurcation increases the complexity and the cost of traffic reassembly, and may also introduce delay jitter at the application layer. Many applications, especially real-time applications, require that their traffic be kept intact, i.e., without de-multiplexing at the source, independent switching at intermediate nodes, and multiplexing at the destination.

In static traffic grooming, the input to the problem includes the logical

topology of the WDM network, relevant network parameters (e.g., the capacity of each lightpath) and a list of requests for data communication. The goal is to determine, if possible, a traffic grooming strategy, so that all the requests for data communication can be handled.

#### 1.4.2 Optimal Regenerator Placement in Translucent Networks

The *Regenerator Placement Problem* (RPP) for translucent networks, identifies a minimum number of nodes in a given network topology which should have 3R regenerating capacity so that any source node can communicate with any destination node [12, 14, 68, 70–73, 86, 95]. This problem is important, since regenerators are expensive and an optimum solution of the problem helps reduce the cost of a translucent WDM network.

In RPP, the input to the problem includes the physical topology of the WDM network, relevant network parameters e.g., the lengths of individual optical fibers connecting two nodes and optical reach for the network. The goal is to determine the minimum number and the locations of the nodes that should have 3R-regeneration capability, so that it is possible to establish a lightpath (either transparent or translucent) between every node-pair in the network.

#### 1.4.3 Optimal Lightpath Allocation in Translucent Networks

Lightpath allocation strategies can be classified into two broad categories. *Dynamic lightpath allocation* is appropriate when the pattern of data communication requests is not known and connections must be set up on demand. *Static lightpath allocation* is used when the data communication requests are

known in advance and do not change significantly over relatively long periods of time. The *Routing with Regenerator Problem* (RRP) uses dynamic lightpath allocation to optimally route a new request for communication using a minimum number of 3R regenerators [4,85,87]. Solving the RRP is important, since each regenerator represents a significant resource of the network and it is useful to set up new lightpaths using minimum resources.

In this thesis we have investigated the following optimal lightpath allocation problems in translucent networks:

- Dynamic lightpath allocation (DLA),
- Dynamic survivable lightpath allocation (DSL A), and
- Static lightpath allocation (SLA).

The DLA and the DSLA runs only after the RPP problem has been solved, so that the locations of the regenerators are known. The goal of DLA is to determine, if possible, a route from the source to the destination of the new request, so that a lightpath (either transparent or translucent) can be established from the source to the destination, and appropriate carrier wavelength(s) can be assigned for the lightpath.

The objective of DSLA is similar to that of DLA except that, instead of finding one route from the source to the destination of the new request, this formulation determines two fiber-disjoint routes from the source to the destination, so that two lightpaths can be deployed (or ready to be deployed) for each request, one is to serve as the primary (working) lightpath, and the other is to serve as the backup (reserved) lightpath, ready to be used if the primary lightpath fails due to any fault (most often due to a link failure resulting from



a broken fiber) in the network.

The input to both the DLA and the DSLA problems include the source and the destination of the new request for communication, the physical topology of the network with relevant network parameters, e.g., the locations of the regenerator nodes, the lengths of the individual optical fibers connecting pairs of nodes, the number of carrier wavelengths supported by each fiber, the optical reach for the network. The input also includes details of all the existing lightpaths in the network, when the new request for communication is processed.

When solving the SLA, the locations of the regenerators are not known. The goal of the SLA problem is to i) identify the locations of the regenerator nodes to minimize the total number of regenerations required, considering all the lightpaths to be deployed, ii) the routes of all the lightpaths through the physical topology, and iii) and assign the carrier wavelength(s) for each lightpath. The input to the static lightpath allocation problem includes the physical topology of the network with relevant network parameters e.g., the lengths of individual optical fibers connecting two nodes, the number of carrier wavelengths supported by each fiber, the optical reach for the network and a list of requests for data communication.

## 1.5 Solution Outline and Contributions

### 1.5.1 A Branch, Price and Cut Approach for Optimal Traffic Grooming

We have studied the problem of finding an optimal solution of the non-bifurcated, static traffic grooming problem. To find an optimal solution, conventionally

the problem is formulated as an Binary Integer Linear Program (BILP). We have used a technique for solving ILP's, called the *Branch, Price and Cut* (BPC) [82] algorithm, using the *arc-chain* representation to solve this problem. We have shown that our approach has the following advantages:

- our formulation for the problem, where we repeatedly solve a LP, uses a significantly smaller basis size, so that each iteration of the simplex process [18] runs faster than formulations based on standard network flow techniques,
- the representation satisfies the requirements of the Generalized Upper Bounding [18], which can be exploited to further expedite the simplex algorithm,
- most of the variables in our LP automatically assume a binary value, so that we have to carry out branching relatively infrequently,
- it is possible to develop a very efficient pricing policy, so that the entering column can be found quickly,
- it is possible to restrict the search space significantly by using additional cutting planes based on the concept of lifting constraints [82].

The net result is an ILP formulation that produces optimal results much faster than conventional formulations. This is clearly supported by the experimental results we have presented in Section 3.5.

### 1.5.2 A Branch and Cut Approach for Optimal Regenerator Placement

To solve the RPP problem, we have presented two Mixed Integer Linear Programs (MILP). The first one is a compact MILP formulation based on standard network flow programming techniques using the node-arc formulation. To the maximum extent possible, we have replaced binary variables by continuous variables, with requisite constraints so that these continuous variables are constrained to have a value of 0 or 1 only. This made the formulation faster. This formulation is simple to implement and can be readily solved by a commercial solver, such as the CPLEX [36] to produce optimal solutions for small and medium size networks within a reasonable amount of time. The second formulation is interesting, since it efficiently solves the RPP problem optimally for relatively larger networks. Here we have used a *branch-and-cut* approach [45] to solve the RPP problem. The formulation has an exponential number of constraints, known only implicitly. However, our experiments reveal that we only need a relatively small number of such constraints, so that the basis size is, in general, quite small and the LP relaxations can be solved very quickly. Experimental results (Section 4.4) clearly show our branch and cut formulation can handle large networks beyond the capacity of conventional MILP's.

### 1.5.3 Optimal Lightpath Allocation

In [67], an important restrictive property of RRP was identified, which was not been taken into account in any earlier algorithm for RRP. We have taken account of this property in our algorithms for DLA and DSLA. We have also

identified some special characteristics of static lightpath provisioning in translucent networks that have not been reported earlier.

### 1.5.3.1 Dynamic Lightpath Allocation

We have presented an ILP formulation to solve DLA in translucent optical networks that works within a reasonable time with small networks. This serves as a benchmark for our second approach for larger networks. Our second approach is based on the A\* algorithm [32], and uses an admissible heuristic. This produces near-optimal solutions in a fraction of the time needed for optimal solutions.

### 1.5.3.2 Dynamic Survivable Lightpath Allocation

We have presented two ILP formulations for DSLA. The formulations are, to some extent, similar to the formulation for DLA. These formulations use the concept of shared path protection [88]. The first one is an formulation that finds the optimal solutions after an exhaustive search. Due to the large number of binary variables, this formulation works only with small networks. For larger networks we have proposed the second formulation which reduces the search space significantly and works much faster. One problem with the second formulation is that, due to the reduced search space, it occasionally fails to find a solution, even when a feasible solution exists.

### 1.5.3.3 Static Lightpath Allocation

We have proposed an ILP formulation for SLA, which finds an optimal solution to the problem. For larger networks we have proposed a two-step heuristic.

The first step of the heuristic uses another ILP to determine a *preliminary* route for each lightpath to be set up. The second step uses a search algorithm, that takes the preliminary route as its input and solves the RWA problem where the actual route used by a lightpath is allowed to differ from the preliminary route, if needed. Our proposed heuristic can produce “near optimal” solutions using only a fraction of the time needed to obtain an optimal solution.

## 1.6 Thesis Organization

The rest of this thesis is organized as follows. In Chapter 2 we have reviewed basic concepts of WDM optical networks, traffic grooming, wide area optical networks, translucent optical networks and some Operations Research (OR) techniques which we have used in our investigation. We have presented our work on optimal traffic grooming in Chapter 3. In Chapter 4 we have presented our research on optimal regenerator placement in translucent optical networks. Chapter 5 include our works on lightpath allocation in translucent networks. We have concluded this thesis, with our suggestions for future works, in Chapter 6.

## Chapter 2

# Review on Related Topics

This chapter reviews topics relevant to the research reported in this thesis including

- *WDM* optical networks
- *Operations Research* (OR) techniques for solving Integer Linear Problems useful for WDM network design.

### 2.1 WDM Optical Networks

A single optical *fiber* has a potential bandwidth of nearly 50 Terabits per second (Tbps), which is about four orders of magnitude higher than the currently achievable electronic processing speed of a few gigabits per second (Gbps) [3]. Because of this limitation of the speed of electronic processing, it is not possible to exploit the entire bandwidth of optical fibers by using only one high capacity optical signal/fiber, and it is desirable to find an effective technology that can efficiently exploit the huge potential bandwidth capacity of optical

fibers. The emergence of *wavelength division multiplexing* (WDM) technology has provided a practical solution to meeting this challenge. With WDM technology, multiple optical signals, using different carrier wavelengths, can be transmitted simultaneously and independently over a single fiber, with each signal carrying data at a rate of a few gigabits per second (Gbps). WDM technology has become popular, since it significantly increases the usable bandwidth of an optical fiber [93]. In WDM optical network, a *lightpath* is defined as an all-optical connection from one end-node to another, used to carry data in the form of encoded optical signals. Such a lightpath always starts from an end-node, traverses a number of fibers and router/end-nodes, and ends in another end-node.

In the following sections we have discussed some of the fundamental optical devices and technologies that are used in WDM optical networks.

### 2.1.1 Optical Fibers

*Optical fibers* are long, thin strands of glass arranged in bundles called *optical cables* and are used to transmit optical signals over long distances. An optical fiber consists of a cylindrical *core* of silica (Figure 2.1), with a higher refractive index, surrounded by cylindrical *cladding*, also of silica, with a lower refractive index [3]. The idea of optical communication using a fiber is that, if an optical signal passing through an optical medium with a higher refractive index, say  $\mu_1$ , meets another optical medium with a lower refractive index, say  $\mu_2$ , at an angle greater than the critical angle  $\sin^{-1}\mu_2/\mu_1$ , total internal reflection takes place where the signal is entirely reflected back into the denser medium [3]. Optical signal propagates through the core of the fiber using a series of such total internal reflections (Figure 2.2).

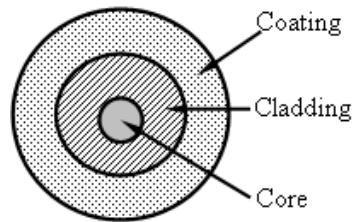


Figure 2.1: Cross-section of an optical fiber

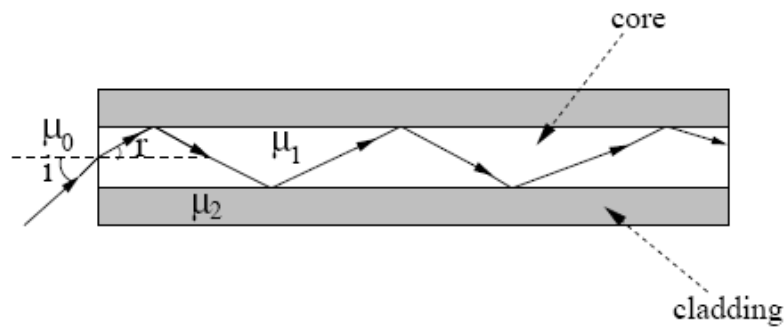


Figure 2.2: Propagation of optical signal using total internal reflection

The data transmission capacity (or bandwidth) of a fiber is strongly dependent on the length and the quality of the fiber. The longer a fiber or the lesser the quality, the lower is the achievable transmission rate [10]. We will discuss later that the transmission rate of an optical fiber also depends on other physical properties of the optical fibers as well as the optical devices used.

### 2.1.2 WDM Technology

Wavelength division multiplexing (WDM) is an optical multiplexing technology used to efficiently exploit the huge bandwidth capacity of the optical fibers. It is conceptually similar to frequency modulation (FM) technique that is being used in radio communication systems for over a century. The basic



principle is to divide the huge bandwidth of an optical fiber into a number of non-overlapping sub-bands or optical *channels*<sup>1</sup> and transmit multiple optical signals simultaneously and independently in different optical channels over a single fiber [3].

The attenuation of an optical signal propagating through a fiber is acceptably low (around 0.2 dB/km) in the wavelength band of 1260 to 1675 nanometers (nm). One is centered at 1300 nanometers (nm) and the other at 1500 nm. Within these intervals, the band from wavelengths 1530 to 1565 nm is called the C-band (conventional band) and is widely being used for optical communication in WDM networks [93].

Using WDM technology, a single optical fiber can carry a number of optical signals. The range of wavelengths allowed for each signal must be non-overlapping. It is convenient to visualize the available bandwidth of an optical fiber (such as the C-band) as a set of ranges of wavelengths or, as they are usually called, channels. Each signal is allotted a distinct channel such that each channel has the sufficient bandwidth to accommodate the modulated signal. In order to avoid any interference between different signals, each channel is separated from its neighboring channels by a certain minimum bandwidth called *channel spacing* (Figure 2.3). Typically, a channel bandwidth of 10 GHz and a channel spacing of 100 GHz are currently being used. This means that the C-band can accommodate up to 80 channels, each having a bandwidth of 10 GHz. Shorter channel spacing (25 GHz) will lead to as many as 200 channels in the C-band alone.

---

<sup>1</sup>The term “channel” is interchangeably used with carrier wavelength in optical network community.

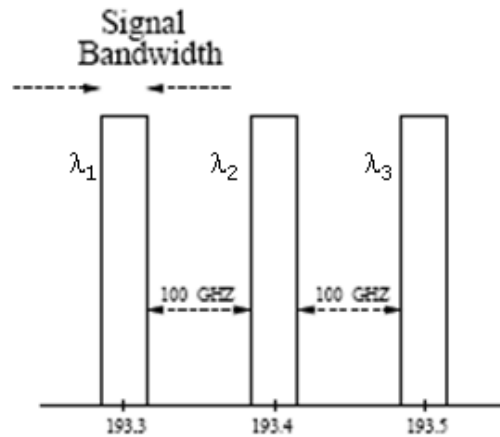


Figure 2.3: Signal bandwidth and channel spacing (modified from [58]).

### 2.1.3 Optical Add-Drop Multiplexers (OADM)

An *optical add-drop multiplexer* (OADM) is a device used for multiplexing and routing different channels of optical signals into or out of an optical fiber in a WDM network system (Figure 2.4). The word “add” here refers to the capability of the device to add one or more new optical signals using an unused channel to an existing set of WDM signals, each using a different channel, and the word “drop” refers to the capability to drop (i.e., remove) one or more optical signals received by the device as inputs.

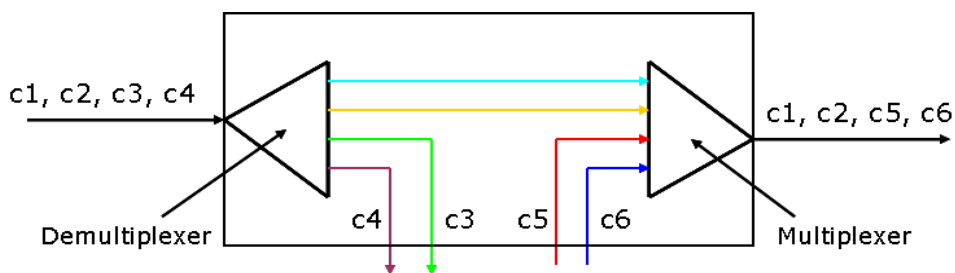


Figure 2.4: Schematic diagram of an OADM

A traditional OADM consists of three stages: an *optical demultiplexer*,

an *optical multiplexer* and a method of reconfiguring the paths between the optical demultiplexer, the optical multiplexer and a set of ports for adding and dropping signals. The optical demultiplexer separates the signals on the input fiber, using different wavelengths and directs them to the optical multiplexer or to the drop ports as it has been configured. The optical multiplexer combines the incoming optical signals, which are not routed to the drop ports, with the signals received at the add ports, onto a single output fiber.

#### 2.1.4 Wavelength Routers ( $\lambda$ Routers)

*Wavelength routers* (Figure 2.5) - which are also called  $\lambda$  (Lambda) routers, or more frequently an *Optical Cross-Connects (OXC)* - are normally positioned at any end-node or network junction points or router nodes. In an OXC, optical signals from an incoming fiber are first demultiplexed, then the demultiplexed wavelengths are switched by optical switching modules. After switching, the optical signals are multiplexed onto an outgoing fiber by optical multiplexers. The whole process is carried out without going through any O-E-O conversion.

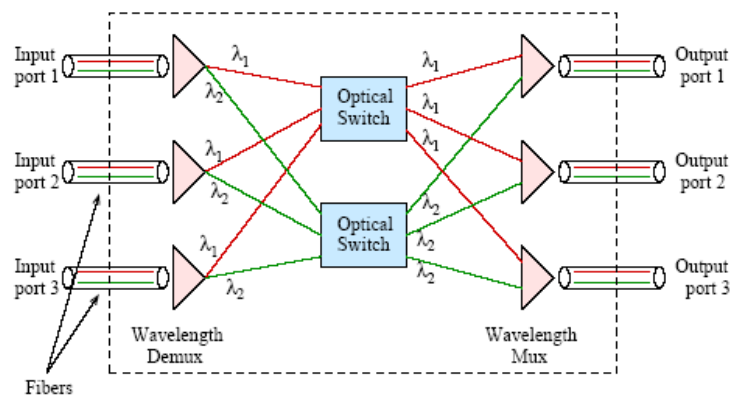


Figure 2.5: Wavelength Router (Optical Cross Connect)

### 2.1.5 Optical Transmitters and Receivers

An *optical transmitter* is a device that accepts an electronic signal as its input, processes this signal, and uses it to modulate an optoelectronic device, such as an LED or an injection laser diode, to produce an optical signal capable of being transmitted via an optical transmission medium [29]. An optical receiver is a device that accepts an optical signal as its input, processes this signal through an electro-optical device to convert it into an electronic signal to be further processed by electronic devices.

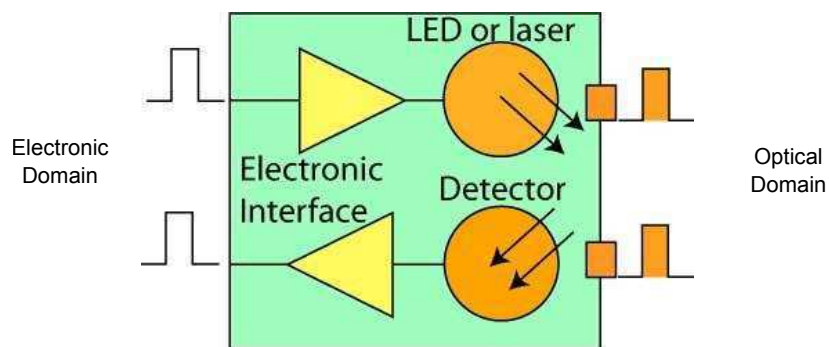


Figure 2.6: Optical Transceivers (modified from [77]).

*Fiber optic transceivers* (Figure 2.6) include both a transmitter and a receiver in the same component. These are arranged in parallel so that they can operate independently of each other. Both the receiver and the transmitter have their own circuitry so that they can handle transmissions in both directions.

### 2.1.6 Wavelength Converters

In general, a lightpath operates on the same channel across all the fiber links it traverses, in which case the lightpath is said to satisfy the *wavelength continuity*

*constraint* [11]. *Wavelength converters* are devices used at the router nodes of WDM or DWDM networks such that a lightpath traveling through multiple fiber links can be assigned different wavelength channels in different links. Using wavelength converters in a network is expensive, but it can eliminate the wavelength continuity constraint of a lightpath so that, in a network with wavelength converters at each node, a lightpath may be assigned different channels on successive fibers used in its route, which greatly reduces number of required channel in a fiber.

### 2.1.7 Physical Topology

The *physical topology* of a network depicts the physical architecture of a network including the relative positions of the end-nodes, router nodes and the optical fibers interconnecting these nodes. An *end-node* in a WDM network is typically a computer in the network where data for communication is either generated or received. A *router node* in a WDM network is a node containing an OXC, that has the capacity to route data from different incoming fibers to different outgoing fibers. A simplified representation of the physical topology of a 4-node network with 3 lightpaths (Figure 2.7(a)) is modeled as a graph (Figure 2.7(b)) where each end node (or router node) in the network is represented as a vertex in the graph and each fiber optic link between two nodes is represented as an edge [59]. Each fiber link is usually bi-directional, so that the graph is undirected.

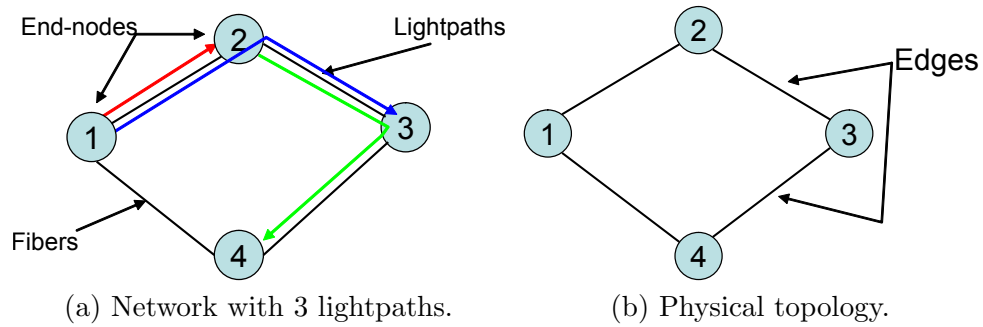


Figure 2.7: A 4-node network and corresponding physical topology.

### 2.1.8 Logical Topology

A logical topology describes the communication capabilities of a WDM network at the optical level and is modeled by a directed graph. Each vertex in the graph corresponds to an end-node in the physical topology and, if there is a lightpath from end-node  $x$  to end-node  $y$ , there is an edge  $x \rightarrow y$  in the graph [59] (also known as a *logical edge*). An example of a logical topology of a 4-node network (Figure 2.7(a)) having 3 lightpaths from node 1 to 2, 2 to 4 and 1 to 3 is shown in Figure 2.8.

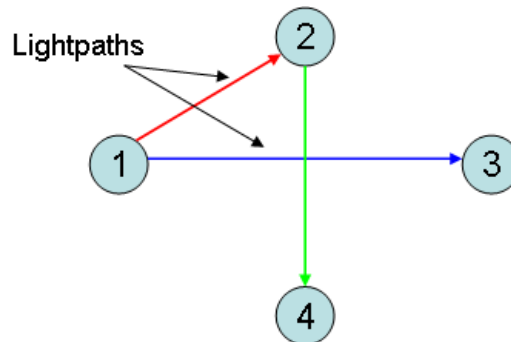


Figure 2.8: Logical topology of a 4-node network.

### 2.1.9 Traffic Grooming

Each lightpath in a WDM network can carry data at a rate between 2.5 Gbps to 10 Gbps, depending on the technology used. Currently, Dense-WDM (DWDM) technology can already achieve up to 320 wavelengths per fiber [83], with each wavelength carrying 40Gbps [5], for a total transmission capacity of up to 12.8 Terabits per sec (Tbps). *Traffic Grooming* in WDM can be defined as a family of techniques for combining a number of low-speed traffic streams from users, so that the high capacity of each lightpath may be used as efficiently as possible. Traffic grooming minimizes the network cost in terms of transceivers and optical switches [3].

Traffic grooming is composed of a rich set of problems, including network planning, topology design, and dynamic circuit provisioning [74]. Traffic grooming strategies can be classified into two broad categories - *static traffic grooming* [97] and *dynamic traffic grooming* [100]. Static grooming is used when the traffic requests are known in advance and the traffic pattern does not change significantly over relatively long periods of time. In this case, it is reasonable to spend a considerable amount of time to determine an optimal grooming strategy. Dynamic grooming is appropriate when the traffic pattern of user requests is not known in advance, and connections must be set up on arrival of a request. The traffic grooming problem based on static traffic demands is essentially an optimization problem. The problem may be viewed differently from different perspectives. One perspective is that, for a given traffic demand, the design has to satisfy all traffic requests as well as minimize the total network cost. The other problem is that, for a given set of resources and traffic demands, maximize the network throughput, i.e., the total amount

of traffic that is successfully carried by the network [92]. In recent years, there has been an increasing amount of research activity on the traffic grooming problem, both in the academia and in the industry.

Traffic grooming can use either the *bifurcated* model or the *non-bifurcated* model [24]. In the non-bifurcated model, each data stream for a user is communicated, using a single logical path from the source of the data stream to its destination. In the bifurcated model, each user data stream is communicated using one or more logical path(s) from the source of the data stream to its destination. In other words, in the non-bifurcated model, whenever there is a user request for communication from a source end-node to a destination end-node, the data stream corresponding to request becomes part of the payload of each lightpath in the selected logical path. This model has been adopted in [26, 33, 88]. In the bifurcated model, the data stream, corresponding to any user request for data communication, is allowed to split into an arbitrary number of data streams at any intermediate point, where the resulting data streams, each having a lower data communication rate than that of the request, is carried by a logical path from source to destination. This process of splitting may occur multiple times as needed. The bifurcated model allows more efficient use of network resources but the non-bifurcated model has a number of technological advantages [88].

When developing a traffic grooming strategy, an important objective is to minimize the amount of network resources used. The *congestion* of a logical topology is defined as the total amount of traffic carried by the lightpath that carries the maximum traffic. A typical objective for non-bifurcated traffic grooming is as follows:



Given a logical topology of a network and a set of requests for data communication, find the route, over the logical topology, for the traffic corresponding to each request for data communication, such that the congestion of the network is as low as possible.

The traffic grooming problem has been extensively studied by the research community. Authors in [16] have developed heuristic algorithms for traffic grooming in unidirectional SONET/WDM ring networks. Algorithms for traffic grooming in WDM networks to reduce the cost of transceivers appear in [42]. A new framework for computing bounds for the problem of traffic grooming in WDM ring topologies is discussed in [24]. The authors in [78] proposed a new capacity correlation model to compute the blocking performance on a multi-hop single wavelength path for grooming traffic in WDM mesh network. [79] studied the performance of dynamic grooming policies for establishing low-rate dependable connections in WDM mesh networks. In [98], the authors proposed a new generic graph model to solve the problem of traffic grooming in heterogeneous WDM mesh networks, using various grooming policies and traffic-request-selection schemes. Various problems faced for grooming dynamic traffics in WDM optical networks to minimize the network cost have been identified in [23]. In [84] the problem of dynamic traffic grooming in WDM mesh networks has been addressed, by first designing a preliminary static logical topology, based on estimated traffic loads, and then routing each client call, arriving dynamically, on the established logical topology. Two exact formulations for employing backup multiplexing and dedicated backup reservation with minimizing the number of primary path sharing a link for enabling traffic grooming capability in the design of survivable WDM mesh

networks have been proposed in [26]. In [35] the authors proposed protection for multi-granular optical networks against near-simultaneous dual-failures using capacity re-provisioning. In his book [74], the author provided a detailed coverage of survivability and traffic grooming in WDM optical networks. Three approaches for grooming a connection request with shared protection based on generic grooming node architecture for survivable traffic grooming have been proposed in [51]. A dependable traffic grooming algorithm implementing shared path protection in WDM mesh networks appears in [83]. Three efficient heuristic grooming algorithms considering both dedicated and shared path protection at the connection level have been discussed in [89]. In [75], the authors proposed a quasi-optimal method to obtain the solution of the non-bifurcated generalized dedicated protection (IGDP) problem. They have formulated the IGDP problem and introduced different SRLG scenarios to serve as the input of the routing problem. A fast and efficient meta-heuristic algorithm, based on Bacterial Evolutionary Algorithm, has also been introduced to find a minimum cost non-bifurcated generalized dedicated protection solution. Based on an auxiliary graph, the authors in [46] have formulated the problem of network virtualization over WDM networks as a MILP to solve both node mapping and link mapping (traffic grooming) optimally. Based on a grooming graph, they have proposed two greedy heuristics to solve the node-mapping and link-mapping sub-problems separately.

#### **2.1.9.1 Traffic Matrix**

Traffic Matrix specifies the amount of traffic (data communication request), using some convenient unit to represent data transmission rates, to be transmitted between each pair of nodes in the network [23]. In general, the data

Table 2.1: An example of a traffic matrix

Nodes	1	2	3	4
1	0	3xOC-3 1xOC-6 1xOC-12	2xOC-6	3xOC-3 1xOC-6 1xOC-24
2	0	0	1xOC-12 1xOC-24	1xOC-3 1xOC-12
3	12xOC-3 1xOC-6	1xOC-24	0	1xOC-3 1xOC-6
4	5xOC-6	1xOC-6 1xOC-12	1xOC-3 1xOC-6 1xOC-24	0

communication rate is expressed in  $OC - n$  notation<sup>2</sup>, where  $OC - 1$  is equal to 51.8 Mbps. If there are  $N$  nodes in a network, the corresponding traffic matrix is an  $N \times N$  matrix and denoted by  $T = \{t_{sd}\}$ , where  $t_{sd}$  is the traffic request from node  $s$  to node  $d$  (Table 2.1).

### 2.1.10 Routing and Wavelength Assignment (RWA)

The logical topology in a WDM optical network is defined using a set of logical edges or lightpaths. To establish a lightpath, it is important to find a suitable route for it in the physical topology and assign a channel to it for every fiber in its route. Given a physical topology and a set of connection requests, the problem of setting up of lightpaths and assigning channels to each of these lightpaths is known as Routing and Wavelength Assignment (RWA) problem [92]. In a network where no wavelength converter is available, a lightpath must

<sup>2</sup>Data communication rate can also be expressed as the fraction of the capacity of a lightpath. It can also be expressed by the actual data rate as  $n$  Mbps or  $n$  Gbps.

be assigned the same channel on all the fiber links it traverses, satisfying the wavelength continuity constraint. In networks with full wavelength converters at each node, the channel used by a lightpath may vary from one fiber to another.

The RWA problem is usually considered under two alternative traffic models. When the set of connection requests is known in advance (for example, given in the form of a traffic matrix) the problem is referred to as *static RWA*, while when the connection requests arrive at random times and are served one by one, the problem is referred to as *dynamic RWA*.

#### 2.1.11 Fault Management in WDM Optical Networks

As WDM optical networks are becoming more and more popular for today's fast telecommunication networks and the internet, the demand for a fault free or fully fault tolerant network system is also increasing. Since a huge amount of data can travel at a tremendous speed through the fibers of the optical networks, even a momentary interruption of any component of the network system can cause the loss of a large amount of data.

As optical networks are being rapidly deployed on a global scale, which involves millions of kilometers of optical fibers and thousands of other network components, protecting a network from different types of faults and failures have become particularly important.

In a WDM network, failure may occur in any component of the network. This includes link failures, node failures, channel failures and/or software failures. Link failure is the most common type of fault where the fiber constituting a link between two nodes in the network does not permit data transmission. Since a single fiber can carry 100 or more lightpaths, and each lightpath can

carry data at the rate of 2.5 Gbps to 10 Gbps, even a brief disruption of this traffic is a serious matter [3].

#### 2.1.11.1 Fault Management Techniques

There are two major techniques that are in use to handle link failures in optical networks:

1. *Protection* based techniques.
2. *Restoration* based techniques.

Protection-based techniques are based on the provisioning of backup paths to recover from a failure [55]. During the period of establishing lightpaths, network resources are kept reserved, such that, when a failure occurs, data can be rerouted around the affected links/lightpaths. In a traditional path protection scheme, if a logical edge is established from node  $i$  to node  $j$ , then resources for two lightpaths are actually reserved. The first one, called the *primary* lightpath, carries the data under normal fault-free conditions and the second one, called the *backup* lightpath, which is link-disjoint with respect to the primary lightpath, carries data only when the primary lightpath fails. Whether only the primary lightpath or both the primary and the backup lightpaths would be established, depends upon the network policy. In case of a network failure, such as a broken link on the primary path from node  $i$  to node  $j$ , the primary lightpath from node  $i$  to node  $j$  is disrupted. In this situation the data from node  $i$  to node  $j$  are sent through the corresponding backup lightpath. Since a primary lightpath and the corresponding backup lightpath are link-disjoint, there will always be a valid lightpath from node  $i$  to node  $j$ , for any single link failure scenario. This approach is more efficient

in respect of response time, but the drawback of this approach is that the resources allocated to the backup paths remain idle, and are wasted under normal conditions.

Restoration-based techniques, on the other hand, dynamically search for the spare capacity in the network to establish new lightpaths in order to restore the affected services after a network failure is detected [56]. There is no allocation of resources for backup paths at design time. Such techniques are more efficient in terms of resource utilization. However, restoration takes longer time than protection to restore services (since backup paths are not known in advance) and there is no guarantee that all the affected lightpaths can be restored. In summary, both protection and restoration schemes require the setting up or the creation of new lightpaths, when a fault is detected.

### 2.1.12 Physical Layer Impairments in Optical Networks

The existence of physical limitations (impairments) of fibers and optical components significantly affect the quality of transmission (QoT) and limit the distance an optical signal can travel before its quality degrades to an unacceptable level. These impairments include amplified spontaneous emission noise (ASE), chromatic dispersion (CD), polarization mode dispersion (PMD), filter concatenation(FC) etc., and they are generally termed as the *linear impairments*.

There are other impairments that also affect the quality of transmission (QoT) and are termed as the *non-linear impairments*, such as self-phase modulation (SPM), cross-phase modulation (XPM), four-wave-mixing (FWM). [17, 58].

Linear impairments affect each of the wavelengths or optical channel in-

dividually. These impairments are dependent upon the quality of the optical devices and the distance along the fibers used by an optical signal. On the other hand, nonlinear impairments cause disturbance and interference between adjacent optical channels [2] in the same optical fiber. These impairments also cause interference between signals using the same channel on different fibers passing through an optical device. The severity of these impairments depend upon the WDM technology used, the number of optical devices it passes through, the proximity of the other channels carrying signals and the number of signals passing through a device using the same channel on different incoming/outputgoing fibers.

Taking into consideration all the linear and nonlinear impairments, when designing WDM networks, considerably complicates the problem. To simplify the problem, the concept of *optical reach* allows designers to consider the linear impairments affecting the quality of transmission (QoT) of an optical signal. The optical reach is defined as the distance an optical signal can travel before its quality degrades below a threshold level due to the effects of all the linear impairments. An optical signal that has to be communicated over a distance more than the optical reach, must be regenerated at some node in its path before it exceeds the optical reach limit.

### 2.1.13 Wide Area Optical Networks

With the evolution of technology, optical networks have become wide area networks reaching almost every corner of the globe. A modern day optical network may have fibers connecting individual nodes which are thousands of kilometers apart. To establish any communication path beyond the optical reach, which typically ranges from 800 to 3000 kilometers [13], it is necessary

to *reamplify*, *reshape* and *retime* the optical signal to restore its quality. These three processes to cleanup and rectify the optical signals are often jointly called as the *3R regeneration* [27].

3R regeneration of optical signal using Optical-Electronic-Optical (O-E-O) conversion is the most popular and mature technique available [86]. The fundamental principle of O-E-O regeneration is to convert an optical signal into the electronic format first, so that the time and shape can be restored, and then use the electronic signal to modulate an optical laser to generate a new optical signal.

Re-amplification is usually done by optical amplifiers. Amplification is not dependent on the bit-rate or the data format, and multiple WDM channels can be amplified simultaneously. However, crosstalk is also amplified and noise is added. WDM channels have to be regenerated individually for the other two regenerations. Re-shaping enables suppression of noise and crosstalk. Re-timing requires an optical clock and a suitable architecture of the regenerator to perform a clocked decision function [64].

Besides the O-E-O technique, it is also possible to carry out 3R regeneration in the all-optical domain without converting an optical signal into an electronic signal. The advantage of all-optical 3R regeneration is its bit-rate transparency, without a bottleneck from electronic modulation. However, the all-optical 3R regeneration technique is not yet mature and is still very expensive [86].

Optical networks may be divided into three basic categories depending upon the concentration and location of 3R regenerating nodes, such as, *opaque* (also called *fully-switched*), *transparent* (also called *all-optical*) and *translucent* networks [54].



### 2.1.13.1 Opaque Networks

In an opaque optical network, a signal from a transmitting node to a receiving node undergoes optical-electronic-optical (O-E-O) conversion to employ 3R regeneration in each intermediate node it passes through [88] (Figure 2.9). Hence in opaque optical network, a single optical hop of a lightpath never spans more than one physical fiber link in the network [88]. Opaque networks with large numbers of nodes tend to be more expensive in terms of time and money as each O-E-O conversion needed for 3R regeneration increases the delay and each 3R regeneration device has a cost associated with it [48, 60].

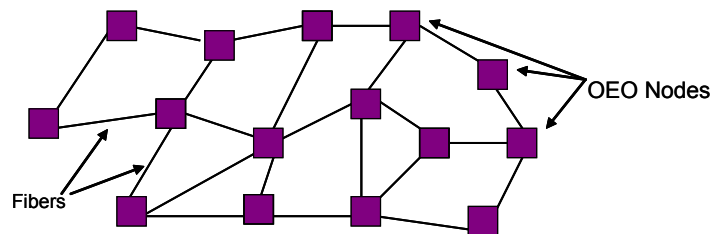


Figure 2.9: An opaque optical network

### 2.1.13.2 Transparent Networks

In a transparent optical network, no signal goes through any O-E-O conversion in any intermediate node in the path from its source to its destination. Signals remain in optical domain from respective source to destination. Though the problems of opaque networks can be overcome in transparent networks, these types of networks are not suitable for today's long haul or ultra long haul networks as optical signals can travel only a limited distance without any type of regeneration (Figure (2.10)).

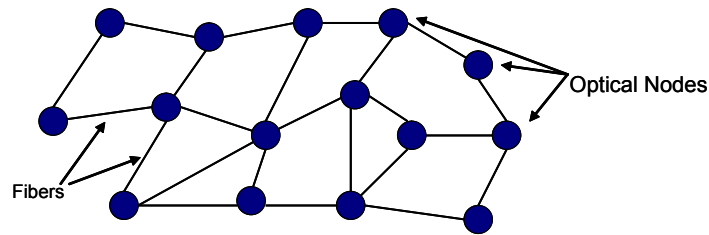


Figure 2.10: A transparent optical network

### 2.1.14 Translucent Networks

To overcome the shortcomings of both transparent and opaque networks, the notion of *translucent networks* was first proposed in [54]. In a translucent network, 3R regenerators are sparsely distributed among a subset of all the nodes of the network (Figure 2.11). Signals are allowed to traverse through the fibers along their respective paths, as far as possible, before its quality falls below a threshold level, at which point only, the signals are subjected to 3R regeneration to restore their quality. This reduces the cost of regenerations as only a few nodes have to have 3R regeneration facilities. On the other hand, a signal can travel a long distance as it may be regenerated when needed, making it possible for the network to be as wide as necessary. In [54] there is a comparative study of the performances of translucent, opaque and transparent networks. It was shown that translucent networks perform better than both of the other types of networks with similar network parameters.

#### 2.1.14.1 Translucent Lightpath

A lightpath in a translucent network [54], that goes through at least one 3R regeneration, is often called a *translucent lightpath*. A translucent lightpath starts from a source node, say  $S$ , uses a path  $S \rightarrow x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_p \rightarrow$

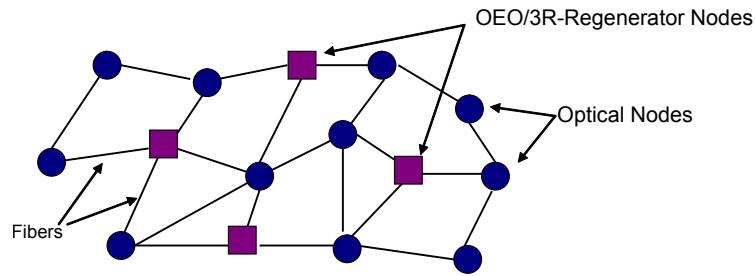


Figure 2.11: A translucent optical network

$R_k \rightarrow \dots \rightarrow R_p \rightarrow \dots \rightarrow R_q \rightarrow \dots \rightarrow D$  and ends at the destination node  $D$ . Each of the paths  $S \rightarrow \dots \rightarrow R_k$ ,  $R_k \rightarrow \dots \rightarrow R_p$ ,  $\dots$ ,  $R_q \rightarrow \dots \rightarrow D$  has a length that does not exceed the optical reach  $r$  and  $R_k, R_p, \dots, R_q$  are regenerator nodes where the incoming signal undergoes 3R regeneration. It is convenient to view a translucent lightpath as a concatenation of components where each component is a transparent lightpath. Each transparent lightpath component is often called a *segment*. A translucent lightpath from  $S$  to  $D$ , using two regenerators  $R_1$  and  $R_2$ , and three segments are shown in Figure 2.12. In general, in a translucent lightpath from  $S$  to  $D$ , the first segment is from  $S$  to some regenerator ( $R_1$  in Figure 2.12), the last segment is from a regenerator ( $R_2$  in Figure 2.12) to  $D$  and the remaining segments are from one regenerator to another (in Figure 2.12, there is only one such segment, from  $R_1$  to  $R_2$ ). We assume that all-optical wavelength converters are not available, so that the wavelength continuity constraint [3] must be satisfied for each segment.

### 2.1.15 Designing Translucent Networks

The following two problems in designing translucent networks have received significant attentions [12, 14, 68, 70–72, 85–88, 90, 95]:

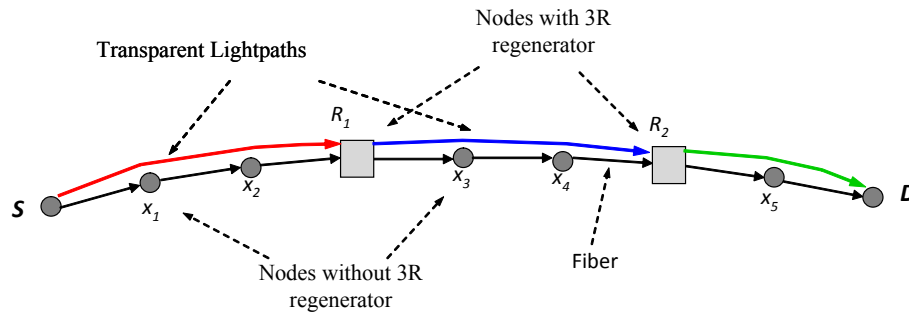


Figure 2.12: A translucent lightpath

- The *Regenerator Placement Problem* (RPP) [12, 14, 68, 70–73, 86, 95]
- The *Routing with Regenerators Problem* (RRP) [4, 85, 87].

The two problems are defined as follows:

**Regenerator Placement Problem:** Given a physical topology, find the minimum number of regenerator nodes and their locations, so that, for each node in the network can communicate with every other node, using either a transparent or a translucent lightpath.

**Routing with Regenerators Problem:** Given a network topology with selected nodes having 3R regeneration capability, and the details of the lightpaths already in existence, find an “optimal route” to establish a lightpath, in response to a new request for communication, using a minimum number of 3R regenerators.

An example of a long haul network with distances between the nodes in km is shown in figure 2.13. If the optical reach is 2000 km, it is clear that an optical signal from node  $A$  cannot reach node  $D$  without 3R-regeneration. However, communication between  $A$  and  $D$  can be established by placing a regenerator either at  $B$  or  $C$ . It may be verified that a minimum of 2 regenerators are

needed for this network and one solution is to place these regenerators at nodes  $B$  and  $D$ . When designing a translucent network, the RPP problem must be solved before tackling the RRP problem.

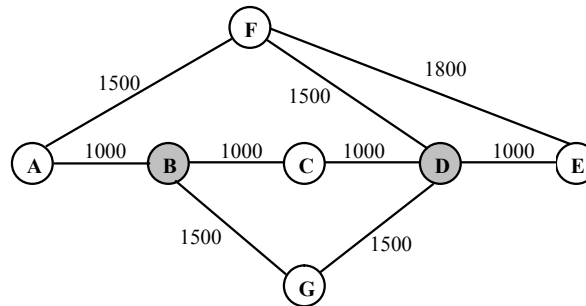


Figure 2.13: Long haul optical network with distances between the nodes in km.

Both the RPP and the RRP are NP-complete [66]. Many researchers have proposed integer linear programs (ILP) for both the problems that can find optimal solutions within a *reasonable* amount time only for networks with relatively small number of end-nodes. To find “near optimal solutions” fairly quickly and for larger networks, researchers have suggested a number of heuristics.

#### 2.1.15.1 Regenerator Placement Problem

Regenerator placement problem has been studied extensively [12, 14, 28, 66–68, 70–73, 86, 95]. In [27] the authors studied two strategies for designing a Translucent Wide Area Network (TWAN). In the first strategy, each node was a potential candidate to have regeneration capability. A lightpath is allowed to traverse as many links as possible until there too much noise accumulated, and a regenerator was allocated to clean up the signal. In the second strategy the network is divided into several regions which they called the *islands of trans-*

*parency*. A lightpath connecting two nodes within an island remains in the optical domain. A lightpath connecting nodes in different islands may or may not go through regeneration at the border nodes (which they call *hub nodes*) depending upon the quality of the signals. In [95] the authors investigate the dimensioning problems for the placement of regenerators in an optical transport network to provide signal-quality guaranteed connections. The authors have developed algorithms to use the regenerators in an efficient manner effectively reducing the network blocking probability. The authors in [65] evaluated the regenerator allocation strategies in order to compensate for signal degradation in all-optical networks. They have pointed out that the transmission impairments are modeled in a cumulative way with respect to path length and the number of traversed nodes. The authors formulated mathematical programs in order to minimize the number of required transponders and fibers. In [47] the authors compared unavailability of port relating to translucent and opaque optical networks.

In [12], the authors have used the *connected dominating set* (CDS) approach to overcome most of the shortcomings of the routing-based approach to the RPP by earlier researchers. The authors of the paper [14] have introduced a new game theoretic formulation for the design and routing of resilient translucent networks that considerably decreases the optimization time and provides near optimal solutions. They have also presented an ILP model to be used as a reference to evaluate the game theoretic algorithm performances. Both of their formulations include pre-calculation of primary and link-disjoint protection paths and take into account the system maximal optical reach distance. In [71], the authors proposed a network dimensioning method that allocates a minimum number of 3R regenerators to optimum locations to build

a cost-effective translucent optical network by combining the advantages of link-based and path-based design approaches.

The authors of the paper [53] proposed a combined approach of regenerator placement based on the estimated signal degradation along the links and nodes and constraint-based routing (CBR) to set up paths according to the demands. Their approach of combined regenerator placement and routing, both based on physical degradation effects, significantly decreases the network blocking probability. The authors of [63] proposed three novel solutions for distributing shared regenerator information. The RPP problem has been identified as an NP-complete problem [67]. A procedure has been outlined in [67], using the concept of *Labeled Connected Dominating Sets* (LCDS), to solve the RPP problem. This approach removes the possibility of an invalid path that could be produced by the procedure in [12]. The authors in [28] presented a theoretical framework to deal with the RPP problems including polynomial time algorithms and approximation algorithms. In [66] The authors have studied both the regenerator placement and routing problems and presented complexity results for them. They have shown that the RPP can be effectively solved using an approximation algorithm for the minimum connected dominated set problem. They have also pointed out that the algorithm presented for the solution of RPP in the paper [28] may sometimes produce invalid solutions. They have shown that the RPP can be formulated as a Minimum Path Concatenation (MPC) problem and have provided the NP-completeness proof of the MPC problem. The paper [15] developed a graph transformation procedure that simplifies the RPP problem equivalent to the maximum leaf spanning tree problem. They have developed a problem reduction procedure, three heuristics, and a post-optimizer for the RPP.

### 2.1.15.2 Routing with Regenerators Problem

Since all-optical wavelength converters are not widely used, it is normally assumed that each segment of a translucent lightpath satisfies the wavelength continuity constraint. The carrier wavelength (or equivalently the channel used) by a translucent wavelength may very well change at the regeneration points, so that different segments constituting a translucent lightpath may have different channel numbers assigned to them.

The authors of [90] studied the Routing and Wavelength Assignment (RWA) problem and considered the need for 3R regeneration when the optical reach for the signal (which they called *transparent length*) is exceeded. The authors proposed a regeneration node selection algorithm called *Max-spare*. This algorithm was compared with a *Greedy algorithm* in conjunction with two routing algorithms, one of which favors the routes with fewer lightpaths sharing the links in the routes, while the other favors the routes with the shorter lengths. In [72] the authors compared two routing strategies considering restoration based network survivability. One of which considers the segment-based restoration scheme, where the transparent path segments between opaque nodes are considered as the entities to be protected. The other strategy considers the conventional schemes where the single spans or entire end-to-end paths are protected. They also presented an algorithm that can determine the placement of minimum number opaque nodes to ensure that every node can communicate with every other node. The authors of the paper [88] addressed the problem of survivable lightpath provisioning in WDM networks, taking into consideration optical-layer protection and some realistic optical signal quality constraints. Their algorithm established a pair of link-disjoint lightpaths for each connec-



tion, given a fixed network topology with a number of sparsely placed O-E-O modules and a set of connection requests. In [85] a suite of dynamic routing schemes were proposed, using dynamic allocation, advertisement and discovery of regeneration resources to support sharing transmitters and receivers between regeneration and access functions. A study of translucent wavelength routed optical network, addressing both the regenerator placement and wavelength routing problems under sparse regeneration in translucent networks appears in [86]. In [94], the authors have adopted the distributed routing policy based on Ant Colony optimized algorithm to efficiently solve the RWA problem in optical networks.

## 2.2 Some Useful Operations Research (OR) Techniques

In general, optimizing optical networks problems are viewed as the *Multi-Commodity Network Flow* (MCNF) problems [1]. Solving MCNF problems typically involves *Integer Linear Programs* (ILP) or *Mixed Integer Linear Programs* (MILP). It is well known that the number of integer variables in a MILP is critically important in determining the time needed to solve the formulation [50]. Most of the MILPs for designing optical networks need binary variables and can find solutions within a reasonable amount of time for relatively smaller networks. For larger networks heuristics are mostly used.

Integer programming, using cutting plane and branch and bound techniques have been investigated very widely within the Operations Research (OR) community [30]. It is recognized that many interesting and practical problems have special structures; and exploitations of such special structures give the maximum opportunity of success with large problems [30]. Unfor-

Unfortunately, the diversity of special structures of these problems is such that procedures that work well on one MILP problem are quite likely to do poorly in another. It has been observed that solving problems involving a very large number of integer variables is often possible by tailoring the algorithm to the special structures of such problems. The following sections include brief overviews of the OR techniques used in the investigations reported in this thesis.

### 2.2.1 Branch and Bound Algorithm

*Branch and bound* (BB or B&B) is a general algorithm for finding optimal solutions of various optimization problems [19, 82]. It consists of a systematic enumeration of all candidate solutions, where large subsets of fruitless candidates are discarded *en masse*, by using upper and lower estimated bounds of the quantity being optimized.

A branch-and-bound algorithm requires two tools [9]. The first one is a splitting procedure that, given a set  $\mathcal{S}$  of candidate solutions, returns two or more smaller sets whose union covers  $\mathcal{S}$ . This step is called *branching*, since its recursive application defines a tree structure (the search tree) whose nodes are the subsets of  $\mathcal{S}$ . When the integer values are binary, the algorithm uses a divide-and-conquer strategy to partition the solution space into two *subproblems* and then recursively solves each subproblem.

Another tool is a procedure that computes the upper and the lower bounds for the minimum (or maximum) value, within a given subset of  $\mathcal{S}$ . This step is called *bounding*. The key idea of the BB algorithm is that, in the case of minimization problem, if the lower bound for some node  $A$ , in the subtree being explored, is greater than the upper bound for the tree, then node  $A$

may be safely discarded from the search. This step is called *pruning*, and is usually implemented by maintaining a global variable  $m$  (shared among all nodes of the tree) that records the minimum upper bound seen among all the subproblems examined so far. Any node whose lower bound is greater than  $m$  can be safely discarded. The reverse is followed for a maximization problem.

Algorithm 1 gives an overview of the Branch and Bound algorithm for the minimization of a Binary Integer Linear Problem (BILP). In this algorithm,  $P^i$  is the current subproblem under consideration.  $\vec{\beta}^*$  is the incumbent optimal solution.  $\vec{\beta}^i$  is the current LP solution value for problem  $P^i$ .  $z^*$  is the global objective value.  $z^i$  is the objective value for the current solution of  $P^i$ .

---

**Algorithm 1** Branch and Bound Algorithm (Minimize)

---

**Input:** Initial problem formulation  $P^0$   
**Output:** Incumbent optimal solution  $\vec{\beta}^*$

- 1:  $List \leftarrow \{P^0\}$
- 2:  $z^* \leftarrow \infty$
- 3: **while** ( $List \neq \emptyset$ ) **do**
- 4:    $P^i \leftarrow \text{choose\_problem\_from\_list}(List)$
- 5:    $(z^i, \vec{\beta}^i) \leftarrow \text{solve\_LP\_relaxation}(P^i)$
- 6:   **if** ( $P^i$  is not feasible) || ( $z^i \geq z^*$ ) **then**
- 7:      $\text{fathom}(P^i)$
- 8:   **else if** (all values in  $\vec{\beta}^i$  are integers) **then**
- 9:      $(z^*, \vec{\beta}^*) \leftarrow (z^i, \vec{\beta}^i)$
- 10:     $\text{fathom}(P^i)$
- 11:   **else**
- 12:      $(P_1^i, P_2^i) \leftarrow \text{branch}(P^i)$
- 13:      $List \leftarrow List \cup \{P_1^i\} \cup \{P_2^i\}$
- 14:   **end if**
- 15: **end while**
- 16: **return**  $\vec{\beta}^*$

---

### 2.2.2 Branch and Price

A *branch and price* algorithm is a branch and bound algorithm and is useful when the number of binary variables in an MILP is very large compared to the number of constraints. In such cases, the columns of the constraints are defined implicitly. While solving the relaxed LP, if a column is not present in the current basis, then the corresponding variable is implicitly taken to have a value of zero. The process of dynamically generating variables whose values should be non-zero is called *pricing*. Hence, LP-based branch and bound algorithms in which the variables are generated dynamically are known as branch and price algorithms [7, 20]. When using the branch and price technique, each LP relaxation is solved initially with only a small subset of the variables present. These variables correspond to the columns in the initial feasible solution. To find an optimal solution the *pricing problem* is solved, to try to identify a column to enter the basis. If such a column is found, the LP is re-optimized. Such columns and corresponding variables may be generated using, for example, the implicit column generation technique in the Dantzig-Wolfe decomposition [21].

### 2.2.3 Branch and Cut

A *branch and Cut* algorithm is also a branch and bound algorithm in which the cutting planes are generated as the branch and bound algorithm proceeds [7]. Here each cutting plane is a valid inequality defined by the requirement of binary variables. In the LP-based branch and bound, efficiency depends substantially on the tightness of the relaxations, that is, how close is the LP-optimal value to the IP-integer optimal value. The LP relaxations can be tightened by adding globally valid inequalities, i.e., those that are valid for the

original feasible set [82]. The goal is to prune the branch and bound tree as much as possible. With additional constraints resulting from the “cut” phase, when evaluating a node in the node queue, the size of the LP to be solved may increase significantly. In other words, there is a trade-off - with many cuts, solving the LP is slower but potentially may lead to solving fewer LP’s.

The method solves the linear program without the integer constraint using the regular *simplex algorithm* [18]. When an optimal solution is obtained, and this solution has a non-integer value for a variable that is supposed to be integer, a cutting plane algorithm is used to find further linear constraints which are satisfied by all feasible integer points but violated by the current fractional solution [50]. If such an inequality is found, it is added to the linear program, such that resolving it will yield a different solution which is hopefully “less fractional”. This process is repeated until either an integer solution is found (which is then known to be optimal) or until no more cutting planes are found.

The function “CPXsetcutcallbackfunc” from CPLEX *callable library* [36] is used to set and modify the user-written callback function to add cuts. The user-written callback is called by CPLEX during MILP branch and cut for every node in the solution tree that has an LP optimal solution with an objective value below the global upper bound but which does not have an integer solution. Cuts that are added at a node remain part of all the subsequent subproblems. Once cuts are added the current subproblem is re-solved and re-evaluated. If the new LP solution is still does not have an integer solution and the objective value is below the upper bound, the cut callback is called again.

### 2.2.4 Branch, Price and Cut

A *branch, price and cut* (BPC) algorithm is a combination of all three algorithms described before. This algorithm is especially useful to solve problems with large number of variables as well as, large number of constraints to satisfy. The algorithm starts with a small set of columns corresponding to the variables as well as, a small set of constraints corresponding to those variables. As the algorithm proceeds, new (entering) columns are added replacing old (leaving) columns. At the same time new valid constraints corresponding to the binary variables are also added.

### 2.2.5 Knapsack Problem

The classic knapsack problem<sup>3</sup> is the unbounded problem of filling a knapsack of volume  $V$  taking items from  $n$  piles of items. For all  $i$ , the  $i^{\text{th}}$  pile of items contains an infinite number of identical items, each with a weight of  $w_i$  and volume  $v_i$ . We are allowed to pick any number of items from each pile without exceeding the total volume  $V$  of the knapsack. The objective is select items so that the total weight of the knapsack, once it is filled, is as much as possible.

The Integer Linear Program formulation for the problem may be written as follows:

$$\text{maximize} \quad w_1 \cdot x_1 + w_2 \cdot x_2 + \dots + w_n \cdot x_n \quad (2.1)$$

subject to

$$v_1 \cdot x_1 + v_2 \cdot x_2 + \dots + v_n \cdot x_n \leq V \quad (2.2)$$

---

<sup>3</sup>Summarized from <http://www.ifors.ms.unimelb.edu.au/tutorial/knapsack/index.html>

$$x_1, x_2, \dots, x_n \in \{0, 1, \dots\} \quad (2.3)$$

Our interest is in the so-called “0-1 knapsack problem” in which the variables have a value of either 0 or 1. In other words, we are allowed to pick at most one item from each pile and the problem is to find which items should be placed in the knapsack so that the weight of the knapsack is as large as possible. To handle this, constraint (2.3) has to be modified as follows:

$$x_1, x_2, \dots, x_n \in \{0, 1\} \quad (2.4)$$

### 2.2.5.1 0-1 Knapsack Inequalities

In order to find out inequalities defined by the requirements of integer variables, 0-1 knapsack inequalities are useful. Given a constraint,  $\sum_{j=1}^n a_j \cdot x_j \leq b$  where  $x_j \in 0, 1$  for all  $j$ , the goal is to find a cut which is “strongest”, in the sense that the inequality gives the most stringent possible restriction on possible combinations of values of  $x_1, x_2, \dots, x_n$ . The following definitions and techniques are taken from [82]. The discussions are for the constraint  $\sum_{j=1}^n a_j \cdot x_j \leq b$  in the original LP and we will use  $N$  to denote the set  $\{1, 2, \dots, n\}$ .

Definition 1:

A set  $C \subseteq N$  is a *cover* if  $\sum_{j \in C} a_j > b$ .

Definition 2:

A set  $C \subseteq N$  is a *minimal cover* if  $C$  is a cover and there is no  $j \in C$  such that  $C - \{j\}$  is a cover.

Definition 3:

If set  $C \subseteq N$  is a cover,  $\sum_{j \in C} x_j \leq |C| - 1$  is a valid inequality.

Starting with a minimal cover  $C$  and the corresponding inequality defined above, details of an algorithm to determine a valid facet defining inequality is available in [82].

### 2.2.6 Arc-chain Representation

Most mathematical formulations using Multi-commodity Network Flow (MCNF) uses the *node-arc representation* [1], where many of the constraints are specified for each node in the network. For many interesting problems (e.g., minimum-cost multi-commodity network flow) the basis size, using the node-arc formulation, can grow rapidly with the size of the network. For a large network, this can become a serious problem, limiting the size of the network that can be handled.

There is an alternate representation for network flow problems in the operations research community called the *arc-chain representation* [3]. A *chain* [8] for a given commodity  $k$  from source  $S^k$  to destination  $D^k$  is defined as a sequence of edges (called arcs in [80]) representing a path  $[(S^k = i_0 \rightarrow i_1), (i_1 \rightarrow i_2), \dots, (i_{p-1} \rightarrow i_p = D^k)]$  in the network from  $S^k$  to  $D^k$ . In this representation, a column in the basis either corresponds to some slack variable or to one of the paths used by some commodity  $k$ .

In a network represented by graph  $G = (N, E)$ , where  $N$  is a set of nodes and  $E$  is the set of edges of the network, a chain may be represented by a vector of  $|E|$  1's and 0's, so that, if the  $i^{th}$  element in the chain is 1 (0), the  $i^{th}$  edge does (does not) appear in the chain, for all  $i \in E$ . In general, in a multi-commodity flow problem, for each commodity  $k \in K$ , there may be many chains in the network since commodity  $k$  may use more than one path



from its source to its destination.

One major advantage of the arc-chain representation is that, for many problems, the basis size, especially for a large network, can be much smaller than the node-arc representation.

To illustrate the concept, following example has been taken from [3] (slightly modified). In this example  $\vec{C}_j^k$  represents the  $j^{\text{th}}$  chain of the  $k^{\text{th}}$  commodity – a vector of 1's and 0's.

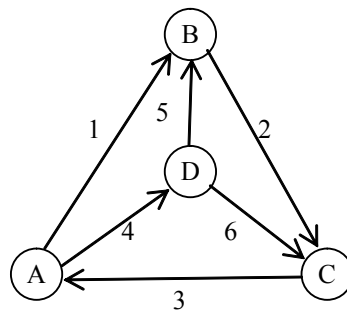


Figure 2.14: A network with four nodes

Figure 2.14 shows a simple network topology with four end nodes and six edges. Each edge is assigned a number from 1 to 6 as shown. This network carries two commodities  $K^1$  and  $K^2$ . The source and destination for commodity  $K^1$  ( $K^2$ ) is  $A$  ( $D$ ) and  $B$  ( $C$ ). There are two paths in this network for commodity  $K^1$ . The first path consists of the edge  $A \rightarrow B$  and the second path is  $A \rightarrow D \rightarrow B$ . The chain  $\vec{C}_1^1$ , representing the first path, is vector  $[1, 0, 0, 0, 0, 0]$  since the edge  $A \rightarrow B$  has been assigned edge number 1. Similarly, the chain  $\vec{C}_2^1$ , corresponding to the second path  $[A \rightarrow D \rightarrow B]$ , is the vector  $[0, 0, 0, 1, 1, 0]$ . Commodity  $K^2$  has two paths so that the chains  $\vec{C}_1^2$  and  $\vec{C}_2^2$  are vectors  $[0, 0, 0, 0, 0, 1]$  and  $[0, 1, 0, 0, 1, 0]$ .

A network having  $m$  edges and  $|K|$  commodities may be represented by an

arc-chain incidence matrix  $AC$ . If there are  $\hat{n}^k$  chains for the  $k^{th}$  commodity,  $\forall k, 1 \leq k \leq |K|$ ,  $AC$  is an  $m \times \hat{n}$  matrix where  $\hat{n} = \hat{n}^1 + \hat{n}^2 + \dots + \hat{n}^{|K|}$  is the total number of chains for all commodities. Each chain of a commodity corresponds to a column in matrix  $AC$  so that the first  $\hat{n}^1$  columns of  $AC$  correspond to chains for commodity  $K^1$ , the next  $\hat{n}^2$  columns correspond to chains for commodity  $K^2$ , and so on.

The arc-chain matrix  $AC$ , in the case of the network shown in Figure 2.14 with two commodities  $K^1$  and  $K^2$  is shown below.

$$AC = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

In this example, the rows of arc-chain matrix  $AC$  corresponds to the edges  $1, 2, \dots, 6$  of the network. Columns 1 and 2 (3 and 4) correspond to the chains for commodity  $K^1$  ( $K^2$ ).

## Chapter 3

# Optimal Traffic Grooming in WDM Networks

### 3.1 Introduction

In WDM optical networks, traffic grooming technique is used to combine a number of low-speed traffic streams from users to send over a single lightpath, so that the high capacity of the lightpath may be used as efficiently as possible. Efficient traffic grooming strategy can not only minimize the network costs in terms of transceivers and optical switches [3], but also can increase the network throughput i.e., the total amount of traffic successfully carried by the network with given set of network resources [92].

Traffic grooming strategies can be classified into two broad categories - *static traffic grooming* [97] and *dynamic traffic grooming* [100]. Static grooming is used when the traffic requests are known in advance and do not change significantly over relatively long periods of time. In this case, it is reasonable to spend a considerable time to determine an optimal grooming strategy. Dy-

dynamic grooming is appropriate when the pattern of user requests is not known and connections must be set up on request. Both the static and the dynamic traffic grooming can be divided into two models. One is known as the bifurcated traffic grooming, where traffic requests are allowed to be split and sent through multiple paths from its source to its destination. The other is called the non-bifurcated traffic grooming, where a traffic request is sent through a single path from source to destination. In this chapter we have presented our work on optimal non-bifurcated traffic grooming in WDM optical network considering the static grooming strategy.

We will use the term *congestion* to denote the maximum of the total amount of data using a particular lightpath. The objective of our algorithm is to determine a non-bifurcated traffic grooming strategy, such that the congestion is as low as possible. The advantages of having a minimum value of congestion in a network, when carrying out traffic grooming, are well known [3]. It is convenient to view the problem of traffic grooming for minimum congestion as a multi-commodity network flow programming (MCNF) [1] problem, where each traffic request between a source-destination pair  $(S^k, D^k)$  for the traffic grooming problem corresponds to a distinct *commodity*  $k$ , to be shipped from the source  $S^k$  to the destination  $D^k$ . Conventionally, to find an optimal solution to this problem, it is formulated as a MCNF problem, specified as an Binary Integer Linear Program (BILP). Solving a BILP with a large number of binary variables is time consuming, since the time required to solve such problems, typically, increases exponentially with the number of binary variables [50].

We propose to use a recent technique, for solving certain ILPs, called Branch, Price and Cut (BPC) [82] to solve our problem. Experimental results (Section 3.5) show that our proposed BPC algorithm can solve larger

instances of the problem, and is significantly faster than solving it with conventional node-arc formulation.

In Section 3.2, we have used a conventional BILP using the node-arc formulation. In Section 3.3 we have proposed an arc-chain formulation. In Section 3.4 we have shown how we may apply BPC, using the arc-chain formulation, to obtain optimal solutions to the non-bifurcated traffic grooming problem more efficiently. In Section 3.5 we have shown how the performances of our BPC formulation compares to the performances using formulation described in Section 3.2.

### 3.2 An ILP for Non-bifurcated Traffic Grooming

In the formulation below we have used the following symbols.  $N$  is the set of all nodes in the network.  $E_L$  is the set of all logical edges (lightpaths) in the network.  $K$  is the set of all commodities or traffic requests.  $\Upsilon^k$  is the amount of traffic for commodity  $k \in K$ .  $F_{ij}^k$  is a binary variable denoting the flow of commodity  $k$  on the logical edge  $(i, j) \in E_L$ . If  $F_{ij}^k = 1(0)$  it means the commodity  $k$  uses (does not use) the lightpath  $(i, j)$ . We have used  $\Lambda_{max}$  to denote the congestion of the network.

The ILP formulation to minimize  $\Lambda_{max}$ , the congestion, when carrying out traffic grooming in WDM networks can be given as:

$$\text{minimize } \Lambda_{max} \tag{3.1}$$

subject to:

$$\sum_{j:(i,j) \in E_L} F_{ij}^k - \sum_{j:(j,i) \in E_L} F_{ji}^k = \begin{cases} 1 & \text{if } i = S^k, \\ -1 & \text{if } i = D^k, \\ 0 & \text{otherwise.} \end{cases} \quad (3.2)$$

Equation (3.2) must be satisfied  $\forall k \in K, \forall i \in N$ .

$$\sum_{k \in K} \Upsilon^k \cdot F_{ij}^k \leq \Lambda_{max}, \forall (i, j), (i, j) \in E_L \quad (3.3)$$

$$F_{ij}^k = \{0, 1\} \quad (3.4)$$

### 3.2.1 Justification for the ILP Formulation

Equation (3.2) is a standard network flow conservation equation. Equation (3.3) corresponds to logical edge  $(i, j)$ , for all edge  $(i, j) \in E_L$  in the logical topology.  $\Upsilon^k$  contributes to the amount of data on logical edge  $(i, j)$ , only if the commodity  $k$  uses the edge  $(i, j)$  (i.e.,  $F_{ij}^k = 1$ ). The left hand side of equation (3.3) is the total flow on edge  $(i, j)$ , considering all the commodities. This total flow on logical edge  $(i, j)$  must be less than or equal to the  $\Lambda_{max}$ . Since the objective is to minimize  $\Lambda_{max}$ , this constraint means that  $\Lambda_{max}$  will be set to the maximum value, considering all edges, of the total amount of data carried by any edge. When the ILP terminates, the flows on different logical edges will be such that the value of  $\Lambda_{max}$  is as low as possible.

Equation (3.4) states that the variable  $F_{ij}^k$  is a binary variable.

The time needed to solve this formulation grows rapidly with the number of nodes in the network. In general, the time complexity of an ILP is  $O(2^N)$ , where  $N$  is the number of binary variables. If the number of end-nodes in a

network is  $n$ , then the potential number of commodities  $K$  is  $O(n^2)$ . Since the number of logical edges from or to an end-node is determined by the number of optical transceivers in the node, the number of logical edges (lightpaths)  $m$  is  $O(n)$ . Therefore, the number of binary variables  $F_{ij}^k$  in the problem is  $O(n^3)$ , which clearly explains how quickly the problem becomes intractable with a modest increase of the number of nodes in the network.

### 3.3 Traffic Grooming using Arc-chain Formulation

Non-bifurcated traffic grooming using the arc-chain formulation may be done as follows:

$$\text{minimize } \Lambda_{max} \quad (3.5)$$

Subject to:

$$\sum_{k \in K} \sum_{j=1}^{n^k} \mu_i^{jk} \cdot u_j^k \leq \Lambda_{max}, \forall i, 1 \leq i \leq m \quad (3.6)$$

$$\sum_{j=1}^{n^k} u_j^k = 1, \forall k \in K. \quad (3.7)$$

$$u_j^k = \{0, 1\}, \forall k \in K, \forall j, 1 \leq j \leq n^k. \quad (3.8)$$

In the above formulation,  $K$  is the set of all commodities.  $n^k$  is the number of all possible chains for commodity  $k$ .  $m$  denotes the number of logical edges in the logical topology. If the  $j^{\text{th}}$  chain for commodity  $k$  includes edge  $i$ , then we define  $\mu_i^{jk} = \Upsilon^k$ , otherwise  $\mu_i^{jk} = 0$ . Binary variable  $u_j^k$  is the flow of

commodity  $k$  using the  $j^{\text{th}}$  chain for this commodity. If commodity  $k$  uses its chain  $j$ ,  $u_j^k = 1$ , otherwise  $u_j^k = 0$ .  $\mu_i^{jk}$  represents the amount of traffic for commodity  $k \in K$  through the logical edge  $i$ ,  $1 \leq i \leq m$ .

Our goal is to minimize the network congestion  $\Lambda_{max}$ . Equation (3.7) and (3.8) ensures that among all  $n^k$  chains, commodity  $k$  uses only one chain implementing non-bifurcated traffic grooming.

It is clear that the total number of columns in the above formulation is  $n^1 + n^2 + \dots + n^{|K|}$ . Since  $n^k$  for any  $k \in K$  may be exponential, the number of binary variables can be exponential too.

### 3.4 Traffic Grooming using Branch, Price and Cut

To solve this traffic grooming problem efficiently, in our Branch, Price and Cut (BPC) algorithm, we first solve the relaxed LP. This amounts to allowing the traffic corresponding to each commodity to split into multiple streams, each using a different path from the source of the commodity to the corresponding destination. In other words, the relaxed LP corresponds to a bifurcated traffic grooming problem by allowing  $u_j^k$  to become a continuous variable. Then as the algorithm proceeds, we gradually impose restrictions on each commodity, to reduce its ability to use multiple paths. Eventually we force each commodity to use a single path from its source to its destination, giving us the desired non-bifurcated traffic grooming solution.

We have given a high-level description of the Branch, Price and Cut (BPC) algorithm in Algorithm 2 below. The idea in Algorithm 2 is to repeatedly apply the bounding phase (which includes solving a LP, using an efficient pricing strategy and a cut-generation step) followed by the branching phase.



Within the bounding phase, in the cut-generation step, we generate *lifting constraints* to represent valid inequalities for the convex hull of integer solutions and add them to the linear description of the current subproblem and its descendants. Branching restricts the number of end-nodes where bifurcation of specified commodities is allowed. Each branch puts restrictions on the edges that can be used by a specified commodity, so that we gradually move from bifurcated traffic grooming to non-bifurcated traffic grooming.

In Algorithm 2,  $LT$  refers to the supplied logical topology over which we have to carry out all data communication.  $LT$  is defined as a set of directed logical edges, each edge  $(s, t)$  denoting a lightpath from  $s$  to  $t$ .  $RDC$  denotes the set of requests for data communication, each request specifying the source and the destination of the request and the data communication rate, using the OC- $n$  notation. To get the process started, we solve the relaxed LP. We need an initial basis  $B_{init}$  giving a feasible solution that is, in general, non-optimal. Our LP solver (described in Section 3.4.1) gives an optimal basis  $B_{opt}$  which, in general, uses bifurcated traffic grooming. Corresponding to each basis  $B_{opt}$ , we can compute a solution using only a single column for each commodity (i.e., using non-bifurcated traffic grooming) which we call  $B_{scs}$ . Each such "single column solution" provides an upper bound to the optimal solution value. We will use  $L(U)$  to denote the lower (upper) bound of the value of the congestion obtained from  $B_{opt}$  ( $B_{scs}$ ).

Algorithm 2 starts with  $B_{init}$ , gets a value of  $L(U)$  from  $B_{opt}$  ( $B_{scs}$ ) after computing the optimal basis, using our LP solver. In general, as we progress using repeated bounding, cut-generation and branching, the lower bound increases and the upper bound decreases. The process terminates when they coincide. The BPC gradually restricts the search space for the formulation,

**Algorithm 2** *BPC for traffic grooming*


---

```

1:  $B_{init} \leftarrow \text{find\_initial\_feasible\_solution}(LT, RDC)$ 
2:  $B_{opt} \leftarrow \text{arc\_chain\_solver}(B_{init})$ 
3:  $(L, U) \leftarrow \text{find\_congestions}(B_{opt})$ 
4: if  $L = U$  then
5:    $\text{exit}(\text{done})$ 
6: end if
7:  $\text{insert\_into\_priority\_queue}(B_{opt})$ 
8:  $\text{solution\_found} \leftarrow \text{FALSE}$ 
9: while  $(\text{priority\_queue} \neq \emptyset \ \& \ \text{solution\_found} = \text{FALSE})$  do
10:   $B_{opt}^0 \leftarrow \text{remove\_first\_from\_priority\_queue}()$ 
11:  if  $(L^0 < U)$  then
12:     $(B^1, B^2) \leftarrow \text{apply\_branching}(B_{opt}^0)$ 
13:    for  $j = 1$  to  $2$  do
14:       $B_{opt}^j \leftarrow \text{arc\_chain\_solver}(B^j)$ 
15:       $(L^j, U^j) \leftarrow \text{find\_congestions}(B_{opt}^j)$ 
16:       $U \leftarrow \text{minimum\_of}(U, U^j)$ 
17:      if  $(L^j = U)$  then
18:         $\text{solution\_found} \leftarrow \text{TRUE}$ 
19:      else if  $(L^j < U)$  then
20:         $\text{insert\_into\_priority\_queue}(B_{opt}^j)$ 
21:      end if
22:    end for
23:  end if
24: end while

```

---

so that when the process terminates, each commodity is carried by exactly one chain. Functions *arc\_chain\_solver* (*apply\_branching*) of Algorithm 2 are described in Section 3.4.1. The remaining functions are straight-forward and self explanatory.

### 3.4.1 An LP for Bifurcated Traffic Grooming

We note that the inequalities in (1.2) for an edge  $(i, j) \in E_L$ :

$$\sum_{k \in K} \Upsilon^k F_{ij}^k \leq \Lambda_{\max}$$

are 0-1 knapsack inequalities. A set  $C$  is called a *cover* if  $\sum_{k \in C} \Upsilon^k > \Lambda_{\max}$ , and is called *minimal* if for any proper subset  $C'$  of  $C$ ,  $\sum_{k \in C'} \Upsilon^k \leq \Lambda_{\max}$ . Such a minimal cover gives rise to a valid “cover inequality”:  $\sum_{k \in K} F_{ij}^k \leq |C| - 1$ . Cover inequalities can be “lifted” to obtain “*lifted cover inequalities*” which, when added to the formulation (1.1) to (1.4), can strengthen it [6, 7].

Let an optimal solution to the relaxed LP for the ILP in Section (1.3), has a commodity, say commodity  $k$ , which uses more than one chain to send the required flow  $\Upsilon^k$ . Then it is easy to reroute flows among these chains so that either, at least one of the edges in these chain is *saturated* (i.e its flow equals  $\Lambda_{\max}$ ), or only one chain can carry the entire flow  $\Upsilon^k$ .

The cover inequalities, arising out of each saturated edge, are then lifted, adapting an approach given in [82]. If a particular traffic grooming strategy has  $H$  saturated edges, we will have  $H$  lifting constraints, where the  $h^{th}$  lifting constraint corresponds to saturated edge  $e_h$  and has the form  $\sum_{k \in K} \xi_h^k \cdot u^k \leq \mathfrak{L}_h$ , where  $u^k$  is the flow of commodity  $k$  on edge  $e_h$ .

The arc-chain formulation given below is an LP that considers all commodities that the network has to support and gives the optimal choice of chains for each commodity and the corresponding flows on each chain so that the total traffic  $\Lambda_{max}$  carried by the logical edge carrying the maximum traffic is as low as possible. In Step 2 of Algorithm 2, the output obtained by the `arc_chain_solver` gives the optimal bifurcated traffic grooming, so that each commodity, in general, is carried by more than one chain.

$$\text{minimize } \Lambda_{max} \quad (3.9)$$

subject to:

$$\sum_{k \in K} \sum_{j=1}^{n^k} \ell_h^{jk} \cdot u_j^k \leq \mathfrak{L}_h, \forall h, 1 \leq h \leq H \quad (3.10)$$

$$\sum_{k \in K} \sum_{j=1}^{n^k} \mu_i^{jk} \cdot u_j^k \leq \Lambda_{max}, \forall i, 1 \leq i \leq m \quad (3.11)$$

$$\sum_{j=1}^{n^k} u_j^k = 1, \forall k \in K \quad (3.12)$$

$$0 \leq u_j^k \leq 1 \quad (3.13)$$

In the above equations,  $\ell_h^{jk}$  refers to the coefficient for lifting constraint  $h$  of the  $j^{th}$  chain for commodity  $k$  and is defined to be  $\ell_h^{jk} = \xi_h^k$  if edge  $e_h$  appears in chain  $j$  of commodity  $k$ ; 0 otherwise.  $\mu_i^{jk}$  represents the entry for commodity  $k$  for edge  $i$  in its  $j^{th}$  chain. The value of  $\mu_i^{jk} = \Upsilon^k$ , if the edge  $i$  is in the  $j^{th}$  chain for commodity  $k$ ; otherwise  $\mu_i^{jk} = 0$ .  $u_j^k$  is the flow of

commodity  $k$  using the  $j^{\text{th}}$  chain for this commodity.

### 3.4.1.1 Justification for the LP Formulation

The objective of the LP is to minimize  $\Lambda_{max}$ , the congestion. Equation (3.10) is to take care of the cutting step of the Branch, Price and Cut (BPC) algorithm using the lifting constraints.

Equation (3.11) corresponds to edge  $i$ , for all edge  $i$ ,  $1 \leq i \leq m$  in the logical topology. Chain  $j$  of commodity  $k$  contributes to the amount of data on edge  $i$  only if edge  $i$  appears in this chain (i.e.,  $\mu_i^{jk} \neq 0$ ). If so,  $\mu_i^{jk} \cdot u_j^k = \Upsilon^k \cdot u_j^k$  is the amount of commodity  $k$  in chain  $j$  that contributes to the total amount of data flowing on edge  $i$ . The left hand side of equation (3.11) is the total flow on edge  $i$ , considering all chains of all commodities. This total flow on logical edge  $i$  must be less than or equal to the  $\Lambda_{max}$ . Since the objective is to minimize  $\Lambda_{max}$ , this constraint means that  $\Lambda_{max}$  will be set to the maximum value, considering all edges, of the total amount of data carried by any edge. When the LP terminates, the flows on different chains will be such that the value of  $\Lambda_{max}$  is as low as possible.

Equation (3.12) corresponds to commodity  $k$ , and specifies that the sum of all the flows on the chains used for commodity  $k$ , must be equal to 1. This equation ensures that the user requirements for commodity  $k$  are all satisfied.

### 3.4.2 Advantages of the Arc-Chain Representation

This formulation has the following important advantages:

1. The arc chain representation of the problem requires a basis of size  $(H + m + |K|)$ . In a  $n$  node network,  $(H + m)$  has size  $O(n)$  and  $|K|$  has

size  $O(n^2)$ , since we expect most, if not all, pairs of end-nodes will have some communication with each other. This may be compared with the basis size of  $O(n^3)$  in the case of a standard node-arc representation using conventional network flow programming [3].

2. This arc chain representation satisfies the structure for the Generalized Upper Bounding [18], so that, after proper permutations of the rows and the columns, the basis  $B$  can be expressed as

$$B = \begin{bmatrix} R & S \\ T & I \end{bmatrix}$$

Matrices  $R, S$  and  $T$  will have sizes  $(H + m) \times (H + m)$ ,  $(H + m) \times |K|$  and  $|K| \times (H + m)$  respectively. Every column of  $T$  has at most one entry of 1 and all remaining entries are 0's and  $I$  denotes an identity matrix of size  $|K| \times |K|$  [3]. We can avoid inverting the whole basis and need only invert a matrix of size  $(H + m)$  at each step of the revised simplex algorithm [18]. This makes the LP much faster, which has a major repercussion on the execution time of our algorithm, where the LP optimization has to be carried out numerous times.

3. At most  $m$  commodities may require multiple chains in the optimized basis for the relaxed LP. Since  $m$ , the number of logical edges is  $O(n)$  and  $|K|$ , the number of commodities is  $O(n^2)$ ,  $m \ll |K|$ . This property is very important, since it establishes that the number of commodities in the relaxed LP requiring multiple chains is very small. This expedites the branching process significantly.

4. This representation makes it easy to apply a branching rule (3.4.4) appropriate for the problem.

### 3.4.3 The Pricing Policy

The pricing policy determines how we obtain an entering column efficiently. As outlined in [18], the standard approach in the revised simplex method when optimizing a LP of the form  $\mathcal{A} \cdot x = b$  is to find, if possible, an entering column by

- computing the simplex multipliers  $\vec{y} = \vec{c}_B B^{-1}$ , where  $\vec{c}_B$  is a vector of cost coefficients and  $B^{-1}$  is the inverse of the basis  $B$ .
- finding, if possible, a column  $\vec{\mathcal{A}}^p$  in the constraints matrix  $\mathcal{A}$ , such that  $\vec{y} \cdot \vec{\mathcal{A}}^p > c_p$ , where  $c_p$  is the cost associated with the column  $p$ .

In the pricing step of BPC, our objective is to identify, if possible, a column  $\vec{\mathcal{A}}^p$ , by examining a relatively small number of possible entering columns. This is particularly important, since the number of possible paths from a given source of a commodity to its corresponding destination is, in general, very large. Each of these paths represents part of a potential entering column.

To handle this problem, it is possible to adapt Tomlin's approach for solving minimum cost multi-commodity flow problems [80]. Instead of explicitly storing the constraints, as done in an LP solver, Tomlin's approach implicitly keeps track of the constraints and generates a chain, only when it is established that the chain should be part of the column entering the basis.

The central idea is that, in each iteration, the algorithm checks only *one potential chain per commodity*. For each commodity, the algorithm checks only

the *shortest* chain from the source to the destination of that commodity, after assigning an appropriate value as the “distance” of each logical edge. Such a shortest chain may be created, on the fly, using, for instance, Dijkstra’s algorithm [1, 22]. The method then checks if the chain satisfies the condition  $\vec{y} \cdot \vec{\mathcal{A}}^p > c_p$  to be part of an entering column. If so, the algorithm creates the entering column using the shortest chain. The details of the algorithm, tailored for the routing problem, are given below.

In this discussions, we will refer to the  $p^{\text{th}}$  column  $\vec{\mathcal{A}}^p$  of the constraints matrix  $\mathcal{A}$ . This constraints matrix will never be explicitly generated or stored. However we will refer to  $\mathcal{A}$  in the following discussions.

We note that, Equation (3.10) ((3.11)) are inequalities and each of the  $H$  ( $m$ ) constraints specified in Equation (3.10) ((3.11)) will have an associated slack variable. We will use  $x_s^h(x_s^{H+i})$  to represent the slack variable corresponding to the  $h^{\text{th}}$  ( $i^{\text{th}}$ ) constraint specified in Equation (3.10) ((3.11)), for all  $h, 1 \leq h \leq H$  ( $i, 1 \leq i \leq m$ ).

Further, we will use  $\rho_1, \rho_2, \dots, \rho_H$  to denote the first  $H$  simplex multipliers (the first  $H$  elements of vector  $\vec{y}$ ), corresponding to the  $H$  *saturated* edges  $e_1, e_2, \dots, e_H$  of the network, for which  $H$  lifting constraints have been determined.

We will use  $\pi_1, \pi_2, \dots, \pi_m$  to denote the next  $m$  simplex multipliers (elements  $H + 1, H + 2, \dots, H + m$  of vector  $\vec{y}$ ), corresponding to the  $m$  logical edges of the network. The remaining  $|K|$  simplex multipliers (elements  $H + m + 1, H + m + 2, \dots, H + m + |K|$  of vector  $\vec{y}$ ), correspond to the  $|K|$  commodities and will be denoted by  $\alpha_1, \alpha_2, \dots, \alpha_{|K|}$ .

**Theorem 1 a)** *If  $\rho_h > 0$ , for any  $h, 1 \leq h \leq H$ , slack variable  $x_s^h$  is a*



candidate to enter the basis.

b) If  $\pi_i > 0$ , for any  $i, 1 \leq i \leq m$ , slack variable  $x_s^{H+i}$  is a candidate to enter the basis.

c) If, for chain  $j, 1 \leq j \leq n^k$  of commodity  $k \in K$ ,  $\sum_{h=1}^H (-\rho_h) \cdot \ell_h^{jk} + \sum_{i=1}^m (-\pi_i) \mu_{ij}^k < \alpha_k$ , then the variable  $u_j^k$ , corresponding to the chain  $\vec{C}_j^k$ , is a candidate to enter the basis.

**Proof:** a) Let  $p$  be the column corresponding to the  $h^{\text{th}}$  slack variable  $x_s^h$ . The cost coefficient for any slack variable is 0, so  $c_p = 0$ , and the vector  $\vec{A}^p$  consists of 0's except in position  $h$ . Therefore, for  $x_s^h$ ,  $\vec{y} \cdot \vec{A}^p - c_p = \rho_h$ . Since  $\rho_h$  is positive,  $x_s^h$  is a candidate to enter the basis.

b) The proof is similar to that given above.

c) Let column  $\vec{A}^p$ , corresponding to chain  $j$  of commodity  $k$ , be a potential entering column. This column has the value  $\ell_h^{jk}$  in position  $h$ , for all  $h, 1 \leq h \leq H$ , the value  $\mu_{ij}^k$  in position  $H+i$ , for all  $i, 1 \leq i \leq m$ . The remaining  $|K|$  positions are all 0's except for a 1 in position  $H+m+k$ . Since the cost of this column  $c_p = 0$ ,  $\vec{y} \cdot \vec{A}^p - c_p = \sum_{h=1}^H \rho_h \cdot \ell_h^{jk} + \sum_{i=1}^m \pi_i \cdot \mu_{ij}^k + \alpha_k$ .

This is a potential entering column if  $\sum_{h=1}^H \rho_h \cdot \ell_h^{jk} + \sum_{i=1}^m \pi_i \cdot \mu_{ij}^k + \alpha_k > 0$ , so

that  $\sum_{h=1}^H (-\rho_h) \cdot \ell_h^{jk} + \sum_{i=1}^m (-\pi_i) \cdot \mu_{ij}^k < \alpha_k$ . ■

Theorem 1 may be used for identifying, if possible, a column to enter the basis. If any  $\rho_h$  or any  $\pi_i$  is positive, the corresponding slack variable can be entered into the basis and the process terminates. Otherwise, we have to test if a chain for commodity  $k$  can be part of an entering column, for some  $k \in K$ .

To do this we first construct, for commodity  $k$ , a digraph  $G_L^k = (N, E_L)$ , where  $N$  is the set of end-nodes of the network and  $E_L$  is the set of all directed edges  $(s, t)$  such that there is a lightpath from  $s$  to  $t$ . The number of edges in  $G_L^k$  is  $m$ . We assign a length  $\sum_{h=1}^H (-\rho_h) \cdot \phi_{hi}^{jk} + (-\pi_i) \cdot \Upsilon^k$  to edge  $i$ , for all edge  $i, 1 \leq i \leq m$ . The value of  $\phi_{hi}^{jk} = \xi_h^k$ , if edge  $i$  is the saturated edge  $e_h$  for which we have determined the  $h^{\text{th}}$  lifting constraint; otherwise  $\phi_{hi}^{jk} = 0$ .

Since all values of  $\rho_h$  and  $\pi_i$  are negative, the lengths of all edges are non-negative. Each path from source  $S^k$  to destination  $D^k$  of commodity  $k$  corresponds to a chain for commodity  $k$ , and hence part of a potential entering column for commodity  $k$ .

**Theorem 2** *If the length of the shortest path from  $S_k$  to  $D_k$  is not less than  $\alpha_k$ , no chain for commodity  $k$  can be part of an entering column.*

**Proof:** One way to characterize the  $j^{\text{th}}$  chain for commodity  $k$  is to define a variable  $\sigma_{ij}^k$  with the value 1(0) if the  $j^{\text{th}}$  chain for commodity  $k$  includes (does not include) the edge  $i$ . Using this notation and the assignment of lengths mentioned above, the length of any path from  $S^k$  to  $D^k$  is  $\sum_{i=1}^m \left( \sum_{h=1}^H (-\rho_h) \cdot \phi_{hi}^{jk} + (-\pi_i) \cdot \Upsilon^k \right) \cdot \sigma_{ij}^k$ . We note that  $\phi_{hi}^{jk} \cdot \sigma_{ij}^k = \xi_h^k$ , if for some edge  $i$ , the edge  $i = e_h$ , so that the chain passes through edge  $e_h$ . Otherwise,  $\phi_{hi}^{jk} \cdot \sigma_{ij}^k = 0$ . It is easy to see that  $\sum_{i=1}^m \left( \sum_{h=1}^H (-\rho_h) \cdot \phi_{hi}^{jk} + (-\pi_i) \cdot \Upsilon^k \right) \cdot \sigma_{ij}^k = \left( \sum_{h=1}^H (-\rho_h) \cdot \ell_h^{jk} + \sum_{i=1}^m (-\pi_i) \mu_{ij}^k \right)$ . If this length is not less than  $\alpha_k$ , this chain does not satisfy the condition given in Theorem 1 and may not be used to create the entering column. If this path is the shortest path from  $S^k$  to  $D^k$ , then the length of all other paths are also not less than  $\alpha_k$ . ■

Since the length of any edge  $i, 1 \leq i \leq m$  is guaranteed to be 0 or positive, finding the shortest path for each commodity can be done very efficiently using well-known techniques, such as Dijkstra's algorithm [1,22]. Let  $S^k$  ( $D^k$ ) be the source (destination) of commodity  $k$  and let  $\varphi(S^k, D^k)$  be the shortest path (for commodity  $k$ ) from  $S^k$  to  $D^k$ . This shortest path  $\varphi(S^k, D^k)$  corresponds to some chain, say the  $j^{\text{th}}$  chain  $\vec{C}_j^k$ . Then, the length of the shortest path is  $= \sum_{i=1}^m \left( \sum_{h=1}^H (-\rho_h) \cdot \phi_{hi}^{jk} + (-\pi_i) \cdot \Upsilon^k \right) \cdot \sigma_{ij}^k$ . If this length is less than  $\alpha_k$ ,  $\vec{C}_j^k$  is a valid chain to enter the basis. In this case, the path  $\varphi(S^k, D^k)$ , found by the shortest-path algorithm, immediately gives the chain  $\vec{C}_j^k$ . We can immediately compute the first  $H + m$  elements in the entering column from  $\varphi(S^k, D^k)$ . The remaining  $|K|$  elements in the entering column are all 0's, except in position  $k$ , where there is a 1.

To summarize, our pricing policy is to first check if a column corresponding to a slack variable can be an entering column. Otherwise, considering each commodity  $k \in K$ , one by one, we apply Theorem 2 to see if we can find an entering column.

The following steps use Theorem 1 to determine a column to enter the basis:

Step 1) Repeat Step 2 for all lifting constraint  $h, 1 \leq h \leq H$ .

Step 2) If  $\rho_h > 0$ , create an entering column corresponding to slack variable  $x_s^h$  and stop.

Step 3) Repeat Step 4 for all edge  $i, 1 \leq i \leq m$ .

Step 4) If  $\pi_i > 0$ , create an entering column corresponding to slack variable  $x_s^{H+i}$  and stop.

Step 5) Repeat Steps 6–8 for all commodity  $k \in K$ .

Step 6) Assign length  $\sum_{h=1}^H (-\rho_h) \cdot \phi_{hi}^{jk} + (-\pi_i) \cdot \Upsilon^k$  to edge  $i$ , in the logical topology for all  $i, 1 \leq i \leq m$ .

Step 7) Find the shortest path  $\varphi(S^k, D^k)$  from  $S^k$  to  $D^k$ .

Step 8) If the length of the path  $\varphi(S^k, D^k) < \alpha_k$ , create the entering column from the path  $\varphi(S^k, D^k)$  of commodity  $k$ , and stop.

Step 9) No entering column exists. Stop.

This algorithm does not include any step to enter the column corresponding to  $\Lambda_{max}$ . The initial feasible solution automatically inserts  $\Lambda_{max}$  in the basis. Once  $\Lambda_{max}$  is in the basis, it is never a candidate for leaving the basis. The process for finding the leaving column involves the standard ratio test of a revised simplex algorithm [18] and is omitted.

#### 3.4.4 The Branching Strategy

We now describe the branching strategy, using the algorithm below:

Step 1) If no commodity uses multiple chains, no further branching is needed.

Step 2) Pick a commodity requiring multiple chains. Let this be commodity  $k$ , for some  $k \in K$ , having source  $S^k$  and destination  $D^k$ . All these chains have the same source  $S^k$  and destination  $D^k$ .

Step 3) Starting from the first node  $S^k$ , examine successive nodes in all the chains for commodity  $k$  and find the last node which is identical for all the chains. Let this node be  $x_p$ . Let  $\mathbb{S}_p$  denote the edges from  $x_p$  and let

$\mathbb{S}_p^1(\mathbb{S}_p^2)$  denote the set of edges  $\{(x_p \rightarrow x^1), (x_p \rightarrow x^2), \dots, (x_p \rightarrow x^r)\}$  ( $\{(x_p \rightarrow x^{r+1}), (x_p \rightarrow x^{r+2}), \dots, (x_p \rightarrow x^{r+s})\}$ ) of  $r$  ( $s$ ) edges from  $x_k$  which are (are not) used by the chains carrying commodity  $k$ .

Step 4) Partition the set of edges in  $\mathbb{S}_p^1$  into sets of edges  $\mathbb{P}_0$  and  $\mathbb{P}_1$ , such that the difference between the total flows of commodity  $k$  on the edges in  $\mathbb{P}_0$  and the total flows of commodity  $k$  on the edges in  $\mathbb{P}_1$  is as small as possible.

Step 5) Include in  $\mathbb{P}_1$  ( $\mathbb{P}_2$ ) half (remaining half) of the edges in  $\mathbb{S}_p^2$ .

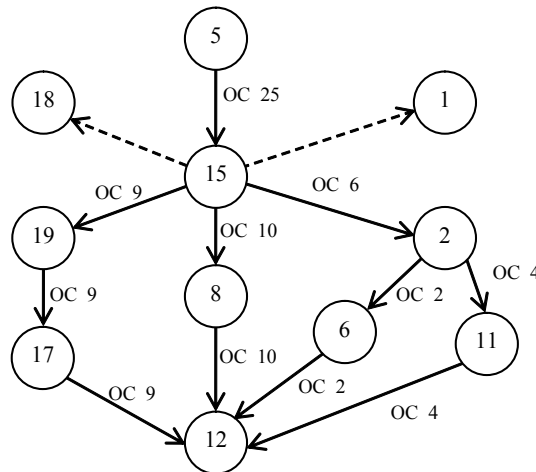


Figure 3.1: An example showing how branching can be used for the traffic grooming problem

**Example:**

In Fig 3.1, commodity  $k$  has a speed requirement of  $OC - 25$ , starts from node 5 and ends at node 12, so that  $S^k$  is 5 and  $D^k$  is 12. For commodity  $k$ , there are four chains using the following paths:

1.  $5 \rightarrow 15 \rightarrow 8 \rightarrow 12$  carrying flow at  $OC - 10$ ,
2.  $5 \rightarrow 15 \rightarrow 2 \rightarrow 6 \rightarrow 12$  carrying flow at  $OC - 2$ ,
3.  $5 \rightarrow 15 \rightarrow 2 \rightarrow 11 \rightarrow 12$  carrying flow at  $OC - 4$ ,
4.  $5 \rightarrow 15 \rightarrow 19 \rightarrow 17 \rightarrow 12$  carrying flow at  $OC - 9$ .

All four chains start from 5 and use the edges  $5 \rightarrow 15$ . Set  $\mathbb{S}_p^1 = \{15 \rightarrow 2, 15 \rightarrow 8, 15 \rightarrow 19\}$  and the total flows on edges  $15 \rightarrow 2, 15 \rightarrow 8, 15 \rightarrow 19$  are  $OC - 6, OC - 10$  and  $OC - 9$  respectively. Since none of these chains use the edges  $15 \rightarrow 1$  and  $15 \rightarrow 18$ , set  $\mathbb{S}_p^2 = \{15 \rightarrow 1, 15 \rightarrow 18\}$ . In this case, after Step 4, set  $\mathbb{P}_0 = \{15 \rightarrow 8\}$  and set  $\mathbb{P}_1 = \{15 \rightarrow 2, 15 \rightarrow 19\}$ . Since there are two edges in  $\mathbb{S}_p^2$ , in Step 5, any one of them may be included in  $\mathbb{P}_0$  and the other one in  $\mathbb{P}_1$  so that one possible solution, after Step 5, is  $\mathbb{P}_0 = \{15 \rightarrow 8, 15 \rightarrow 18\}$  and set  $\mathbb{P}_1 = \{15 \rightarrow 2, 15 \rightarrow 19, 15 \rightarrow 1\}$ .

The branching strategy defines  $\mathcal{S}$  to be the set of all possible non-bifurcated traffic grooming solutions and  $\mathcal{S}_0(\mathcal{S}_1)$  to be the set of possible non-bifurcated traffic grooming solutions where the single chain for commodity  $k$  uses one of the edges in set  $\mathbb{P}_0$  ( $\mathbb{P}_1$ ).

### 3.5 Experimental Results

In our experiments we find that, as the size of the problem (as measured by the number of nodes and the number of requests for data communication)

grows, the time to solve an ILP for the non-bifurcated traffic grooming problem grows rapidly and the problem quickly becomes intractable. We have shown that BPC can solve this problem quite efficiently.

For our experiments, we have considered three well-known networks - the 14 node NSFNET backbone network, the 21 node ARPA-2 network and the 24 node USANET backbone network [69]. We have randomly generated logical topologies for each of these networks. The numbers of transmitters and receivers at each node were randomly selected from 2 to 4.

We set the total numbers of traffic requests to 1000, 1500, 2000, 2500 and 3000 for each of the networks. For a particular network, when considering a specific value of the total number of traffic requests, we randomly assigned the requests to different node-pairs in the network with uniform probability. The size of the requests varied from OC-1 to OC-24 considering OC- $n$  notation used for data communication rate [3]. For a specific value of the total number of traffic requests, we randomly generated 10 sets of requests. The results presented in this section represent the average values of the 10 sets of experiments.

The main objective of our experiments was to calculate the execution time required by our BPC algorithm to optimally route all the requests over the network, such that the overall network congestion is minimized using the non-bifurcated traffic grooming. We have compared the results of our algorithm with that of standard node-arc formulation, solved using CPLEX version 11.1 [36]. We have implemented our BPC algorithm using C programming language and ran the program on a Sun Fire X2200 M2 Server [76].

Figure 3.2 compares the execution time needed using BPC and the time required using the CPLEX solver for different traffic sizes in NSFNET network.

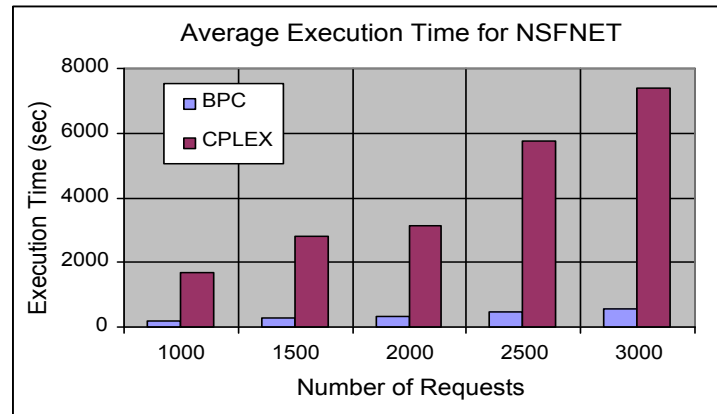


Figure 3.2: Comparison of execution time required by BPC and CPLEX on different size requests.

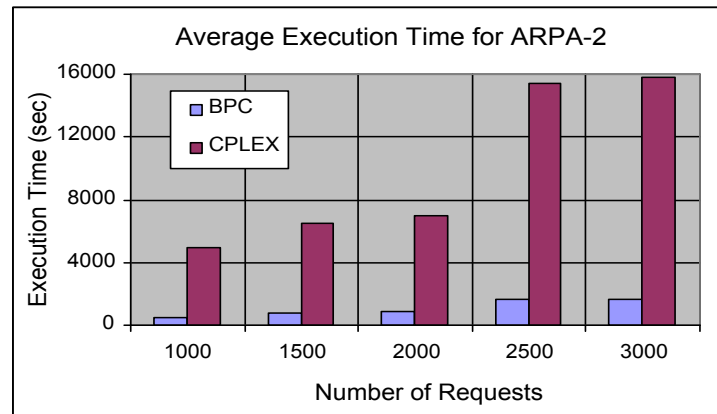


Figure 3.3: Comparison of execution time required by BPC and CPLEX on different size requests.

Figure 3.3 compares the execution time needed for ARPA-2 network, and Figure 3.4 compares the execution time needed for USANET network.

Figure 3.5 shows the graphs representing the percentage of execution time needed by BPC over CPLEX for all the three networks. The figure clearly shows that, for all the cases we have considered, BPC only took, on an average, between 10% to less than 20% of time required by the CPLEX. It also shows



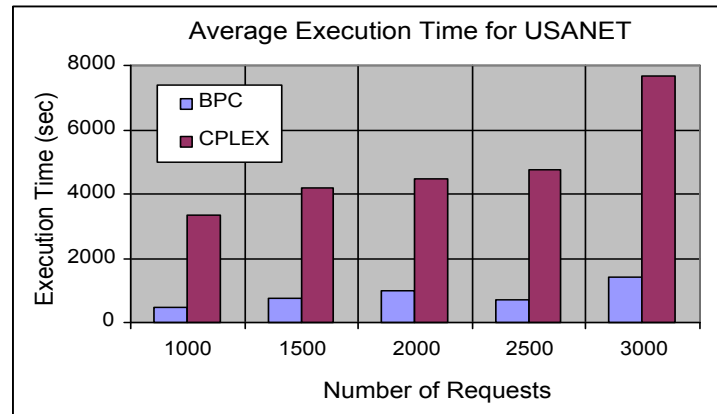


Figure 3.4: Comparison of execution time required by BPC and CPLEX on different size requests.

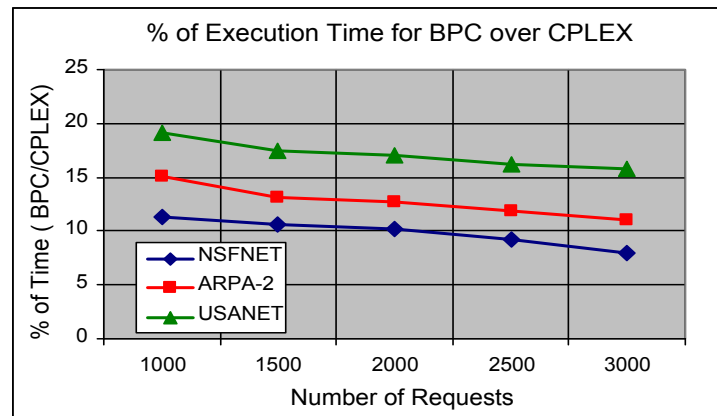


Figure 3.5: Percentage of execution time required by BPC over CPLEX for different size of requests and for different networks.

that, for each network, as the number of requests increases, BPC becomes more and more efficient compared to the CPLEX.

The figures clearly indicates that our algorithm outperforms the CPLEX solver in all the cases. As expected, the execution time using CPLEX increases exponentially with the increase of the number of traffic requests as well as the size of the network, whereas the increase of execution time using our algorithm

is nearly linear. In fact, we have identified one case with 24 node USANET network and only 1000 requests, where the CPLEX solver failed to produce any solution after running for more than 24 hours, at which point we decided to stop the process and considered it as a failure. With the same data set our algorithm produced a solution with optimal congestion value in just 5215.01 sec (approx. 87 min). In all the cases where both BPC and CPLEX could solve the problems, they produced the same optimal congestion values for the networks.

## Chapter 4

# Optimal Regenerator Placement in Translucent Networks

### 4.1 Introduction

In a translucent optical network (reviewed in Section 2.1.14), in order to enable every node-pair to communicate with each other, a selected number of nodes need to be capable of 3R-regeneration (reviewed in Section 2.1.13) of incoming optical signals. The problem of identifying a minimum number of nodes in a given network which should have 3R regenerating capacity, so that each node can communicate with any other node in the network, is known in the literature as the Regenerator Placement Problem (RPP).

To model an optical network we use a graph  $G = (N, E)$ , where  $N$  represents the set of nodes of the network and  $E$  represents the set of edges, where each edge  $(i, j) \in E$  represents a bi-directional fiber between node  $i$  and node  $j$ . Each edge  $(i, j) \in E$  also has a label  $d_{ij}$ , denoting the length of the fiber from  $i$  to  $j$ .

We may view the RPP problem as a network flow problem where we treat the problem of communicating from a source  $S$  to a destination  $D$  as a problem of shipping a distinct commodity from  $S$  to  $D$ . We start by considering every possible ordered source-destination pair  $(S, D)$  in the network and our goal is to send one unit of the commodity corresponding to  $(S, D)$  from  $S$  to  $D$ . If the minimum distance from node  $S$  to node  $D$  is less than or equal to the optical reach  $r$ , no regeneration is needed for the commodity. In other words, the issue of regenerator placement is irrelevant for such commodities and, in order to reduce the size of the problem, our discussions below assume that such commodities are not included while solving the RPP problem. Furthermore, if we can find a route for a commodity from  $S$  to  $D$ , then the reverse of the same route may be used for the commodity from  $D$  to  $S$ . Therefore we may simplify the RPP problem further by considering only the commodity for node pair  $(S, D)$ , if  $S < D$ .

For each commodity we need to consider, our formulations have to select a path and place regenerators on such path, so that

- the path for the commodity has at least one regenerator,
- the distance from the source of the commodity to the first regenerator node in the path  $\leq r$ ,
- the distance from the last regenerator node in the path to the destination of the commodity  $\leq r$ ,
- the distance from any regenerator node to the next regenerator in the path  $\leq r$ .

After the RPP has been solved and the regenerators are in place, given any

source-destination pair  $(S, D)$ , there exists at least one route from  $S$  to  $D$ , so that a viable optical connection can be established from  $S$  to  $D$ , using this route, which may involve 0 or more 3R regenerator nodes.

In this chapter we present two Mixed Integer Linear Programs (MILP) to optimally solve the RPP for translucent networks. The first formulation (FRM-1) is based on standard network flow programming techniques. This formulation can be readily solved by any commercially available mathematical solver, such as the CPLEX [36], and works for relatively small networks (networks having up to 35 nodes). The second formulation (FRM-2) is much stronger, since it efficiently solves the RPP problem for relatively large networks (we have solved 140 node networks in less than 2000 seconds, on an average). We have proposed a *branch-and-cut* (reviewed in Section 2.2.3) approach to solve this problem. The formulation has an exponential number of constraints, known only implicitly. However, our experiments reveal that we only need a relatively small number of such constraints, so that the basis size is, in general, quite small and the LP relaxations can be solved very quickly.

The concept of the *reachability graph* [66] is useful in our formulations. We construct the reachability graph  $G_R = (N, E_R)$  from the graph  $G = (N, E)$  representing a network, by defining  $E_R$  as follows. An edge  $(S, D)$  is in  $E_R$ , iff the shortest distance from  $S$  to  $D$  does not exceed the optical reach  $r$ . A node-pair  $\{(S, D) : S < D\}$ , such that there is no edge from  $S$  to  $D$  in the reachability graph, is a commodity that we have to consider in solving RPP. It is known that the minimum connected dominating set (MCDS) of the nodes in the reachability graph  $G_R$  gives the smallest number of regenerators needed for the graph  $G$  [66].

The rest of the chapter is organized as follows. In Section 4.2 we have pre-

sented a compact node-arc formulation. In Section 4.3 we have presented our branch and cut formulation. In Section 4.4 we have presented the experimental results using the formulations FRM-1 and FRM-2.

## 4.2 A Compact Formulation to Solve the RPP

In this section we present FRM-1, a Mixed Integer Linear Program (MILP) formulation to optimally solve the RPP problem using Multi Commodity Network Flow (MCNF) techniques using node-arc representation.

The idea used in this formulation is to define flow-balance constraints [1] for each commodity of interest and determine a path for each commodity in the reachability graph. We use a regenerator at each intermediate node in the path for the commodity. The objective of the formulation is to minimize the number of nodes with 3R regeneration capability and can be given as:

$$\text{minimize } \sum_{j \in N} \beta_j \quad (4.1)$$

subject to:

$$\sum_{j:(i,j) \in E_R} f_{ij}^k - \sum_{j:(j,i) \in E_R} f_{ji}^k = \begin{cases} 1 & \text{if } i = S^k, \\ -1 & \text{if } i = D^k, \\ 0 & \text{otherwise.} \end{cases} \quad (4.2)$$

Equation (4.2) must be satisfied  $\forall k \in K, \forall i \in N$ .

$$\beta_j \geq \sum_{i:(i,j) \in E_R} f_{ij}^k : \quad \forall k \in K, \forall j \in N | j \neq D^k. \quad (4.3)$$

$$\beta_j = \{0, 1\} : \forall j \in N. \quad (4.4)$$

In the above formulation  $\beta_j$  is a binary variable for each node  $j \in N$  in the physical topology. If the node  $j$  is chosen to be a regenerator node,  $\beta_j = 1$ , otherwise  $\beta_j = 0$ .  $K$  denotes the set of all commodities that we need to consider.  $f_{ij}^k$  represents the flow for commodity  $k \in K$  on edge  $(i, j) \in E_R$  so that, if the commodity  $k$  (or a part of it) uses the edge  $(i, j)$ ,  $f_{ij}^k > 0$ , otherwise  $f_{ij}^k = 0$ .  $f_{ij}^k$  is a continuous variable.

#### 4.2.1 Justification for the Compact Formulation

Equation (4.2) is a standard network flow conservation equation. Equation (4.3) ensures that a commodity  $k \in K$  can only use an edge  $(i, j) \in E_R$  in the reachability graph if, either the node  $j$  is its destination node ( $j = D^k$ ), or the node  $j$  is a regenerator node ( $\beta_j = 1$ ). Since our objective is to minimize the number of regenerator nodes, equations (4.2), (4.3) and the objective function means that the formulation finds a path (or a number of paths) for each commodity, such that the total number of intermediate nodes, considering all the paths used by all the commodities, is minimum. Clearly this solves the RPP.

We used continuous variables  $f_{ij}^k$  for the flow variables. This reduces the number of integer variables and decreases the time needed to solve the formulation. We note the implication that we are allowing the use of multiple paths to communicate each commodity from its source to its destination. We can select any one of the paths used by commodity  $k \in K$  and discard all the other paths without increasing the number of regenerators in the network.

This formulation can be directly given to any commercially available mathematical optimization tool like CPLEX for finding an optimal solution and can generate a solution within a reasonable time with the small and medium sized networks (networks having up to 35 nodes).

### 4.3 RPP using Branch and Cut Algorithm

To find optimal solutions for the RPP problems for large translucent networks, in this section we propose a *branch-and-cut* algorithm. This formulation has, in general, an exponential number of constraints, known only implicitly. For that reason, this formulation cannot be solved directly by CPLEX. However, the formulation may be implemented by interacting with the CPLEX solver, using the control callbacks from the CPLEX callable library [36].

To explain our proposed formulation let us consider followings. As we have mentioned in section 4.1 that we consider only those communication requests as a commodity where the lightpath between the nodes needs at least one regeneration. Let us define a set of all nodes as the *Disconnecting Set of Nodes (DSN)* out of which at least one node must be a regenerator node to establish communication between a given source node and a destination node.

**Definition 1** A set  $\mathbb{D} \subset N$  is a disconnecting set of nodes (DSN), if at least one of the nodes in  $\mathbb{D}$  must be made a regenerator node for a feasible solution of the RPP problem.

For a given network, there could be an exponential number of such disconnecting set of nodes (DSNs).

**Example:** Let us consider the reachability graph for a six node network



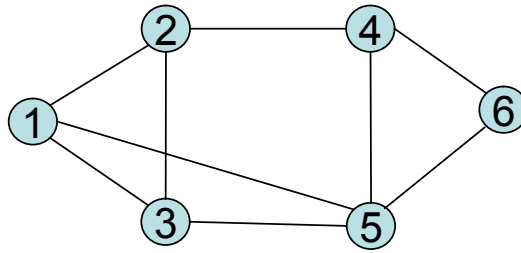


Figure 4.1: Reachability graph for a 6 node network

as shown in the Figure 4.1. Following our discussion in Section 4.1 the commodities that need to be considered are:  $K^1 = (1,4)$ ,  $K^2 = (1,6)$ ,  $K^3 = (2,5)$ ,  $K^4 = (2,6)$ ,  $K^5 = (3,4)$  and  $K^6 = (3,6)$ . Some disconnecting sets of nodes (DSN) are:

- $\{2, 3, 5\}$
- $\{1, 3, 4\}$
- $\{1, 2, 5\}$
- $\{4, 5, 6\}$
- $\{1, 3, 4, 6\}$
- $\{4, 5\}$

#### 4.3.1 An ILP Formulation to Solve the RPP

As before, we assume that we are given a reachability graph  $G_R = (N, E_R)$ . Let  $K$  be the set of commodities we need to consider according to Section 4.1, where each commodity is specified by the corresponding source-destination pair. RPP can be solved using the following binary linear integer program (BILP):

$$\text{minimize } \sum_{j \in N} \beta_j \quad (4.5)$$

subject to:

$$\sum_{j \in \mathbb{D}} \beta_j \geq 1 : \quad \forall \mathbb{D} \in \Omega \quad (4.6)$$

$$\beta_j = \{0, 1\} : \quad j \in N \quad (4.7)$$

In this formulation  $\beta_j$  is, as before, a binary variable corresponding to node  $j \in N$ . If node  $j$  is selected to be a regenerator node,  $\beta_j = 1$ , otherwise  $\beta_j = 0$ .  $\mathbb{D}$  is defined to be a disconnecting set of nodes (DSN).  $\Omega$  is the set of all DSN's.

As before, our objective is to minimize the number of regenerator nodes. Equation (4.6) ensures that at least one node in each DSN  $\mathbb{D}$  must be a regenerator node. Equation (4.6) must be satisfied  $\forall \mathbb{D} \in \Omega$ .

**Lemma 1** *Any solution satisfying constraints (4.6) and (4.7) is a feasible solution for the regenerator placement problem.*

**Proof:** Let there be a solution that satisfies all the constraints of the above BILP but is not a feasible solution for the RPP. In other words, at least one pair of nodes  $(S, D)$  can not communicate with each other, when we have regenerators placed at node  $i$ , if the value of  $\beta_i = 1$  in the current solution. Starting with node  $s$ , we can identify the set  $R$  of all nodes  $v$ , such that  $S$  can communicate with  $v$ . Clearly  $D \notin R$ . Let  $\mathbb{D}$  be the set of all nodes  $i$  in  $R$ , which currently have  $\beta_i = 0$ . In order that  $S$  may communicate with  $D$ , one or more of the nodes in  $\mathbb{D}$  must be equipped with a regenerator. Clearly  $\mathbb{D}$  is a DSN and must have been included in constraint (4.6). This is a contradiction.

■

Since the number of elements in  $\Omega$  is exponential, this formulation has an exponential number of constraints of type (4.6). Specifying such a formulation to CPLEX for any non-trivial network, or solving a large integer problem with an exponential number of constraints, is, in general, not feasible.

As pointed out in Section 4.2, when the RPP is solved, for each commodity of interest, all interior nodes in the selected path from the source to the destination in the reachability graph must be regenerator nodes. We now transform the problem, so that, instead of paths through the reachability graph, we consider paths through a capacitated [1], directed graph, which we will call the *extended reachability graph*. Our objective is that, instead of saying that the path from the source to the destination should pass through regenerator nodes (ie., is allowed to pass through node  $i$  if  $\beta_i = 1$ ), we want to say that the path should pass through edges with some appropriate capacity. Given a reachability graph  $G_R$  and the values of  $\beta_i, i \in N, \beta_i \in \{0, 1\}$ , we define the extended reachability graph  $G_R^X = (V_R^X, E_R^X)$  as follows:

- for each node  $u \in N$ , two nodes  $u^1, u^2$  in  $V_R^X$ ,
- for each edge  $u, v \in E_R$ , two directed arcs  $u^2 \rightarrow v^1, v^2 \rightarrow u^1$  in  $E_R^X$ , each having an infinite capacity,
- for each node  $u \in N$ , a directed arc  $u^1 \dashrightarrow u^2$  in  $E_R^X$ , with a capacity of  $\beta_u$ .

We will use  $\rightarrow$  and  $\dashrightarrow$  to distinguish between these two types of directed arcs.

As an example, Figure 4.2.b shows the extended reachability graph corresponding to the reachability graph shown in Figure 4.2.a.

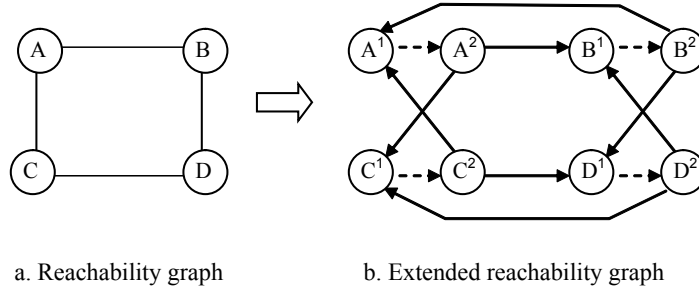


Figure 4.2: Extended reachability graph created from reachability graph

If commodity  $k$  has a route  $S^k \rightarrow a_1 \rightarrow a_2 \rightarrow \dots \rightarrow a_p \rightarrow D^k$  in the reachability graph  $G_R$ , then commodity  $k$  has a route  $S^{k2} \rightarrow a_1^1 \dashrightarrow a_1^2 \rightarrow a_2^1 \dashrightarrow a_2^2 \rightarrow \dots \rightarrow a_p^1 \dashrightarrow a_p^2 \rightarrow D^{k1}$  in the extended reachability graph  $G_R^X$ . Similarly, given a path through the extended reachability graph, it is easy to find the corresponding path through the reachability graph. The flow of commodity  $k$ , on the path  $S^{k2} \rightarrow a_1^1 \dashrightarrow a_1^2 \rightarrow a_2^1 \dashrightarrow a_2^2 \rightarrow \dots \rightarrow a_p^1 \dashrightarrow a_p^2 \rightarrow D^{k1}$ , is limited by  $\text{minimum}(\beta_{a_1}, \beta_{a_2}, \dots, \beta_{a_p})$ . Since all  $\beta$  values are binary, the value of  $\text{minimum}(\beta_{a_1}, \beta_{a_2}, \dots, \beta_{a_p})$  is either 0 or 1. If this value is 0, it means that there cannot be any flow from  $S^{k2}$  to  $D^{k1}$  in  $G_R^X$  using the route  $S^{k2} \rightarrow a_1^1 \dashrightarrow a_1^2 \rightarrow a_2^1 \dashrightarrow a_2^2 \rightarrow \dots \rightarrow a_p^1 \dashrightarrow a_p^2 \rightarrow D^{k1}$ . In other words, in  $G_R$ , all interior nodes in the route  $S^k \rightarrow a_1 \rightarrow a_2 \rightarrow \dots \rightarrow a_p \rightarrow D^k$  from  $S^k$  to  $D^k$  are not regenerator nodes.

**Lemma 2** *In a valid solution for the RPP, any cut [1] in  $G_R^X$  for any commodity must have a capacity of 1 or more.*

**Proof:** When RPP is solved, let a cut in  $G_R^X$  for commodity  $k$  have a capacity of less than 1. Since all  $\beta$  values are binary when RPP is solved, the capacity

of this cut is 0, and hence the maximum flow for this commodity is 0. This means there is no route from the source  $S^k$  to the destination  $D^k$ , where all interior nodes are regenerator nodes. Thus  $S^k$  cannot communicate with  $D^k$ , contradicting the statement that RPP is solved. ■

We note that the converse (i.e., if all cuts have a capacity of 1 or more, we have an optimal solution to the RPP) is not true. If all cuts have a capacity of 1 or more, we have a situation where every node can communicate with every other node but the total number of regenerator nodes is, in general, not minimum.

To lay the foundations for the branch and cut scheme outlined in Section 4.3.2 below, we now consider a situation where we solve a BILP,  $F^{new}$ , with the same objective function as the BILP above where we *have not* included all the DSN's in  $\Omega$ , when constructing the constraints. It is quite possible that the solution to  $F^{new}$  will not correspond to a solution to the RPP, since some constraints are missing. We consider a situation where for some commodity  $k$ , node  $S^k$  cannot communicate with  $D^k$ . This means that there is no path from  $S^k$  to  $D^k$ , where all interior nodes are regenerator nodes.

**Lemma 3** *Node  $S^k$  cannot communicate with  $D^k$ , for some commodity  $k$ , iff the min-cut in  $G_R^X$  for commodity  $k$  has a capacity less than 1.*

**Proof:** The “if” part of the lemma is simply a restatement of Lemma 2 and the proof follows directly. Suppose  $S^k$  can not communicate with  $D^k$ , then there is no path from  $S^k$  to  $D^k$  with all interior nodes as regenerator nodes. Hence max-flow from  $S^k$  to  $D^k = 0$ . Result follows, since max-flow equals to min-cut. ■

**Lemma 4** *The set of nodes  $\{i : (S^k, i) \in E_R\}$  is a DSN.*

**Lemma 5** *If the max-flow for commodity  $k$  in graph  $G_R^X$  is less than 1, the min-cut in  $G_R^X$  for commodity  $k$  identifies a DSN.*

We note that if all commodities have a min-cut of 1 or more, it means that, even though all DSN's are not included in forming the constraints, we have a valid solution for the RPP problem. This is a crucial observation that we have used below.

### 4.3.2 A branch and Cut Scheme to Solve the RPP

The branch-and-cut scheme given below gives an efficient way to solve RPP problem. It is based on the standard branch and bound algorithm [44], summarized in Section 2.2.3, that essentially solves a series of relaxed LP subproblems [36].

The idea is that we start the process by specifying an ILP with the same objective function we used in Section 4.3.1, but with constraints corresponding to a small number of DSN's that may be generated quickly. CPLEX solves the LP corresponding to the ILP using its normal procedure, by relaxing the binary variables  $\beta_i, i \in N$ , so that they have continuous values, as part of the branch and bound algorithm (described in Section 2.2.1). The callback feature of CPLEX allows us to interrupt the branch and bound process, when some events of interest take place. Using this feature, before branching (Step 13 of Algorithm 1 given in Section 2.2.1) we take control to add additional cuts. After including these additional constraints, CPLEX resumes and solves the LP once again. We carry out this process of adding constraints (called "adding user cuts" using callback functions from CPLEX callable library).

This process is repeated as long as we discover new DSN's. Algorithm 3 gives an overview of our branch and cut process.

---

**Algorithm 3** Branch and Cut Algorithm for the RPP
 

---

**Input:** Initial problem formulation  $P^0$   
**Output:** Incumbent optimal solution  $\vec{\beta}^*$

- 1:  $List \leftarrow \{P^0\}$
- 2:  $z^* \leftarrow \infty$
- 3: **while** ( $List \neq \emptyset$ ) **do**
- 4:    $P^i \leftarrow \text{choose\_problem\_from\_list}(List)$
- 5:    $notDone \leftarrow true$
- 6:    $fathom\_current\_problem \leftarrow false$
- 7:   **while** ( $notDone$ ) **do**
- 8:      $(z^i, \vec{\beta}^i) \leftarrow \text{solve\_LP\_relaxation}(P^i)$
- 9:     **if** ( $P^i$  is not feasible) || ( $z^i \geq z^*$ ) **then**
- 10:        $fathom\_current\_problem \leftarrow true$
- 11:        $notDone \leftarrow false$
- 12:     **else if** (all values in  $\vec{\beta}^i$  are integers) **then**
- 13:       **if** ( $\text{valid\_RPP}(\vec{\beta}^i)$ ) **then**
- 14:           $(z^*, \vec{\beta}^*) \leftarrow (z^i, \vec{\beta}^i)$
- 15:       **else if** (violated cuts available with  $\vec{\beta}^i$ ) **then**
- 16:          add user cuts to  $P^i$
- 17:       **else**
- 18:           $fathom\_current\_problem \leftarrow true$
- 19:           $notDone \leftarrow false$
- 20:       **end if**
- 21:     **else if** (violated cuts available with  $\vec{\beta}^i$ ) **then**
- 22:       add user cuts to  $P^i$
- 23:     **else**
- 24:        $notDone \leftarrow false$
- 25:     **end if**
- 26:   **end while**
- 27:   **if** ( $fathom\_current\_problem = false$ ) **then**
- 28:      $(P_1^i, P_2^i) \leftarrow \text{branch}(P^i)$
- 29:      $List \leftarrow List \cup \{P_1^i\} \cup \{P_2^i\}$
- 30:   **end if**
- 31: **end while**
- 32: **return**  $\vec{\beta}^*$

---

Algorithm 3 runs until it either finds an optimal RPP solution, or the

list of all candidate subproblems is exhausted. The algorithm keeps track of the best objective value  $z^*$  found so far and  $\vec{\beta}^*$ , the corresponding values of the variables. To start the process, our program finds an initial problem formulation ( $P^0$ ) and gives it to CPLEX in Step 1. This formulation is the same as that in Section 4.3.1 and contains one DSN for each commodity  $k \in K$  of interest, computed using Lemma 4.

We carry out the branch and cut algorithm using the *while* loop (Step 3 to Step 31). In these steps, CPLEX solves all the candidate subproblems until the solution tree is empty and corresponds to the steps 3 - 15 of Algorithm 1 in Section 2.2.1. Following its normal procedure, CPLEX chooses a subproblem  $P^i$  to solve, from the list of candidate subproblems (Step 4). Steps 7 - 26 denotes another *while* loop. In Step 8 CPLEX finds the objective value  $z^i$ , and the values of the variables  $\vec{\beta}^i$ , corresponding to the solution of the relaxed LP. In Steps 9-25, we have to consider the following cases:

1. If the LP is infeasible or its objective value is greater than or equal to the best objective value  $z^*$  found so far, CPLEX discards the formulation  $P^i$  by setting the variable *fathom\_current\_problem* equal to *true* and exits the *while* loop.
2. If the solution to  $P^i$  happens to be an integer solution (i.e., all values in  $\vec{\beta}^i$  are integers), we check, using Lemma 3, if the solution is a valid RPP solution (Step 13). If it is a valid RPP solution, we have found a better solution than the best we found so far and we replace the best objective value (incumbent values)  $z^*(\vec{\beta}^*)$  found so far by the current objective value (incumbent values)  $z^i(\vec{\beta}^i)$  (Step 14).
3. If the integer solution is not a valid RPP solution, this means that,



at least for one  $k \in K$ ,  $S^k$  cannot communicate with  $D^k$ . For this case, the max-flow from  $S^k$  to  $D^k$  is less than 1. Lemma 5 indicates that the corresponding min-cut gives us a DSN that is not satisfied by the current solution. For all commodities  $k \in K$ , such that  $S^k$  cannot communicate with  $D^k$ , we add the DSN generated using Lemma 5 in Step 16. If no cut is found CPLEX discards the formulation by setting *fathom\_current\_problem* equal to *true* and exit the *while* loop.

4. If, on the other hand, the solution to  $P^i$  is a fractional solution (i.e., at least one value in  $\vec{\beta}^i$  is not an integer), using the same CPLEX callback feature for adding user cuts, our program checks if any violated cut is available for any commodity with the current values in  $\vec{\beta}^i$  (Step 21).

A simple way to identify the violated cuts is to apply Lemma 5 and find a min-cut for each commodity. If the capacity of this cut is less than 1, the DSN corresponding to the minimum cut is not satisfied by  $\vec{\beta}^i$ . Any polynomial time algorithm for *max-flow* [1] can find these violated cuts quite efficiently.

If cuts are found, we add them to the formulation (Step 22) and CPLEX goes back to the beginning of the *while* loop (in Steps 7-26) and solves the LP again. If no cut is found, CPLEX exits this *while* loop.

Once out of this inner *while* loop, CPLEX checks the value of *fathom\_current\_problem* (Step 27). If it is true, it abandons further processing of problem  $P^i$  (called *fathoming* in the OR literature). Otherwise, CPLEX splits the current problem  $P^i$  into two subproblems  $P_1^i$  and  $P_2^i$  (Step 28) and adds both the subproblems to *List* (Step 29). We repeat Steps 3 - 31 as long as there are candidate subproblem(s) in *List*. Once *List* is empty, CPLEX returns with the incumbent

optimal solution  $\vec{\beta}^*$ , which is the optimal RPP solution.

**Example:** Let us consider the reachability graph for the six node network shown in Figure 4.1. Following our discussion in Section 4.1 the commodities that need to be considered are:  $K^1 = (1,4)$ ,  $K^2 = (1,6)$ ,  $K^3 = (2,5)$ ,  $K^4 = (2,6)$ ,  $K^5 = (3,4)$ ,  $K^6 = (3,6)$ , and  $K^7 = (5,6)$ .

To start the process, we find a small set of constraints, one for each commodity using Lemma 4. From the graph we can easily identify those constraints as:

- $\beta_2 + \beta_3 + \beta_5 \geq 1$ , (DSN for  $K^1$ )
- $\beta_2 + \beta_3 + \beta_5 \geq 1$ , (DSN for  $K^2$ )
- $\beta_1 + \beta_3 + \beta_4 \geq 1$ , (DSN for  $K^3$ )
- $\beta_1 + \beta_3 + \beta_4 \geq 1$ , (DSN for  $K^4$ )
- $\beta_1 + \beta_2 + \beta_5 \geq 1$ , (DSN for  $K^5$ )
- $\beta_1 + \beta_2 + \beta_5 \geq 1$ , (DSN for  $K^6$ )

The redundant constraints from the above problem can be removed without any loss of generality:

- $\beta_2 + \beta_3 + \beta_5 \geq 1$  (DSN for  $K^1$  and  $K^2$ )
- $\beta_1 + \beta_3 + \beta_4 \geq 1$  (DSN for  $K^3$  and  $K^4$ )
- $\beta_1 + \beta_2 + \beta_5 \geq 1$  (DSN for  $K^5$  and  $K^6$ )

One integer solution of this problem is  $\beta_2 = 1$  and  $\beta_3 = 1$ . But this solution is not a valid RPP solution, as the max-flow for commodities  $K^2, K^4, K^6$  and

$K^7$  are all 0's. To get a valid RPP solution we must find some violated cuts and add them to the current problem and solve the problem once again. Using max-flow algorithm we find the following additional cuts:

- $\beta_4 + \beta_5 \geq 1$ , (DSN for  $K^2$ )
- $\beta_1 + \beta_4 + \beta_5 \geq 1$ , (DSN for  $K^4$  and  $K^6$ )

Adding these additional cuts the problem becomes:

- $\beta_2 + \beta_3 + \beta_5 \geq 1$
- $\beta_1 + \beta_3 + \beta_4 \geq 1$
- $\beta_1 + \beta_2 + \beta_5 \geq 1$
- $\beta_4 + \beta_5 \geq 1$
- $\beta_1 + \beta_4 + \beta_5 \geq 1$

If we ignore the redundant (or loose) constraints, the new problem becomes:

- $\beta_2 + \beta_3 + \beta_5 \geq 1$
- $\beta_1 + \beta_3 + \beta_4 \geq 1$
- $\beta_1 + \beta_2 + \beta_5 \geq 1$
- $\beta_4 + \beta_5 \geq 1$

One optimal integer solution for this problem is  $\beta_2 = 1$  and  $\beta_4 = 1$ , which is a valid RPP solution.

## 4.4 Experimental Results

For our experiments, we have considered different randomly generated topologies. For comparing the performance of our formulation FRM-2 using the Branch-and-Cut approach (Section 4.3), with that of our compact formulation FRM-1 (Section 4.2), we have generated a number of 15, 20, 25, 30 and 35 node networks, and have run both formulations<sup>1</sup>. For evaluating the performance of the formulation FRM-2 for medium and large sized networks, we have generated networks with 40, 60, 80, 100, 120 and 140 nodes. For a given size of the network, we have generated 3 categories of networks, which we have called the *low*, *medium* and *high* “density” networks. We have measured the density of a network by the average number of edges for each node. For networks containing 60 nodes or less, for low density (medium density, high density) networks, we have randomly chosen the degree of each node to lie in the range from 2 to 3 (3 to 5, and 4 to 7 respectively). For networks containing 80 nodes and more, we have randomly chosen the degree of each node to lie in the range 4 to 5, 5 to 7, and 6 to 9 for a low, a medium and a high density network respectively.

If there is an edge between nodes  $x$  and  $y$ , we have randomly selected the length of the fiber connecting  $x$  and  $y$  to lie in the range 800 to 2800 km. We have selected the optical reach  $r$  to be 3000 km.

For our first set of experiments to compare the performances of FRM-1 and FRM-2, for each category of networks with sizes from 15-node to 35-node, we have randomly generated 10 sets of physical topologies and the results reported in this paper represent the average values of those 10 sets of physical topologies.

---

<sup>1</sup>We were limited to networks with 35 or fewer nodes, since the formulation FRM-1 takes an unacceptable amount of time to solve if the network has more than 35 nodes.

For our second set of experiments to evaluate the performance of FRM-2, for each category of networks with sizes from 40-node to 140-node, we have randomly generated 5 sets of physical topologies and the results reported in this paper represent the average values of those 5 sets of physical topologies. We have carried out the experiments on a Sun Fire X2200 M2 Server machine [76].

We have presented two tables (Table (4.1) and Table (4.2)), as well as two graphs (Figure (4.3) and Figure (4.4)) to demonstrate our experimental results. In column 3 and 4 of both tables, we have shown, respectively, the average number of edges ( $|E|$ ) in the physical topology and the average number of commodities ( $|K|$ ), for different network densities.

Table (4.1) illustrates, along with other data, a comparison of the average execution time (given in seconds) needed to solve the RPP problem using the formulations FRM-1 and FRM-2 (column 7 and column 8, respectively), for different network sizes and for different network densities. The results show that the formulation FRM-1 needs considerably more time than formulation FRM-2 to solve a given problem - ranging, on an average, from approximately 9 times more for 15-node, low-density network to 2150 times more for 35-node, high-density network. The dramatic improvement in the execution time of FRM-2, compared to FRM-1, particularly for larger networks, is remarkable.

Table (4.2) illustrates, along with other data, a comparison of the average execution time (given in seconds) needed to solve the RPP problem using the formulation FRM-2 (column 7) for different network sizes and for different network densities.

The average number of regenerators, determined using the heuristic in [31] and shown in column 5 of Table 4.1 and Table 4.2, is more than the optimum value shown in column 6 of Table 4.1 and Table 4.2, in all the cases. This shows

Table 4.1: Comparing average execution time using FRM-1 and FRM-2

Network Size	Network Density	Average # of		Average # of Regenerators		Average Solution Time (sec)	
		Edges	Commodities	Heuristic	Optimal	FRM-1	FRM-2
15	Low	38.0	79.0	6.4	6.1	0.18	0.02
	Medium	59.0	57.8	3.1	2.8	0.14	0.01
	High	78.6	35.2	2.0	1.9	0.08	0.01
20	Low	49.4	155.5	9.1	8.8	1.33	0.11
	Medium	78.4	125.7	4.5	4.0	1.10	0.03
	High	96.6	94.5	2.7	2.2	0.67	0.02
25	Low	62.2	254.1	11.5	11.0	32.22	0.26
	Medium	95.8	219.1	6.5	5.3	11.77	0.07
	High	134.2	171.7	3.5	3.0	6.78	0.02
30	Low	75.2	379.5	14.0	12.8	158.29	0.47
	Medium	119.8	325.7	6.6	5.7	139.10	0.16
	High	162.0	272.2	4.5	3.8	39.68	0.05
35	Low	87.8	530.5	16.1	15.1	515.62	1.66
	Medium	137.8	473.6	7.7	6.6	427.19	0.27
	High	185.4	405.6	5.8	4.6	430.35	0.20

the need to use an optimal algorithm, rather than a heuristic, to minimize the cost of regenerators.

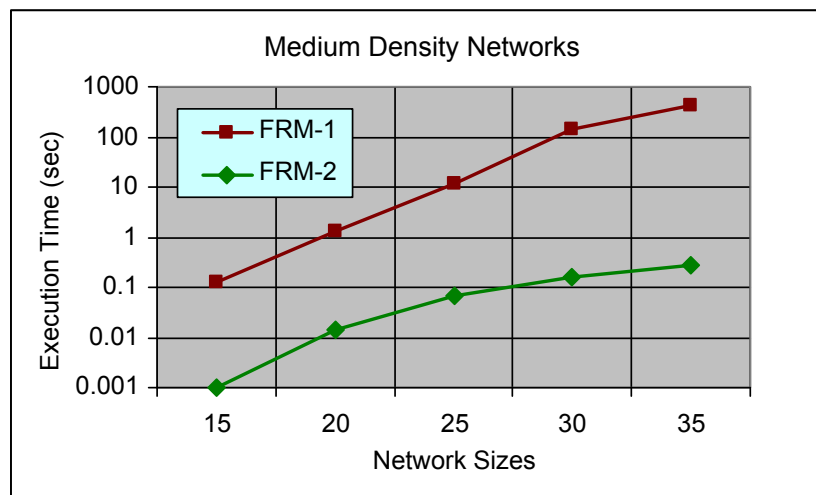


Figure 4.3: Comparing average execution time for medium density networks

Figure (4.3) shows the comparison of average execution time using FRM-1

Table 4.2: Average execution time for large networks using FRM-2

Network Size	Network Density	Average # of		Average # of Regenerators		Average Solution Time (sec)
		Edges	Commodities	Heuristic	Optimal	FRM-2
40	Low	100.0	709.4	18.8	17.4	3.32
	Medium	159.6	640.4	9.2	7.8	0.61
	High	218.0	565.6	6.8	5.2	0.13
60	Low	151.6	1655.0	29.0	27.4	70.27
	Medium	244.0	1558.0	13.8	11.0	9.95
	High	323.6	1398.8	8.4	6.6	1.28
80	Low	359.2	2825.0	16.6	13.4	232.58
	Medium	478.8	2595.6	11.0	8.4	35.73
	High	589.2	2407.4	8.0	6.4	4.51
100	Low	450.8	4505.2	21.4	15.8	582.72
	Medium	592.0	4256.0	13.4	11.0	77.73
	High	743.2	4003.8	11.4	7.8	11.50
120	Low	770.4	4505.2	15.0	11.8	740.17
	Medium	957.2	4256.0	11.0	8.2	96.58
	High	1142.4	4003.8	9.0	6.4	15.94
140	Low	905.6	4505.2	18.4	11.8	1853.53
	Medium	1117.6	4256.0	13.8	8.2	224.74
	High	1328.8	4003.8	10.4	6.4	44.33

and FRM-2 for different sized networks with medium density.

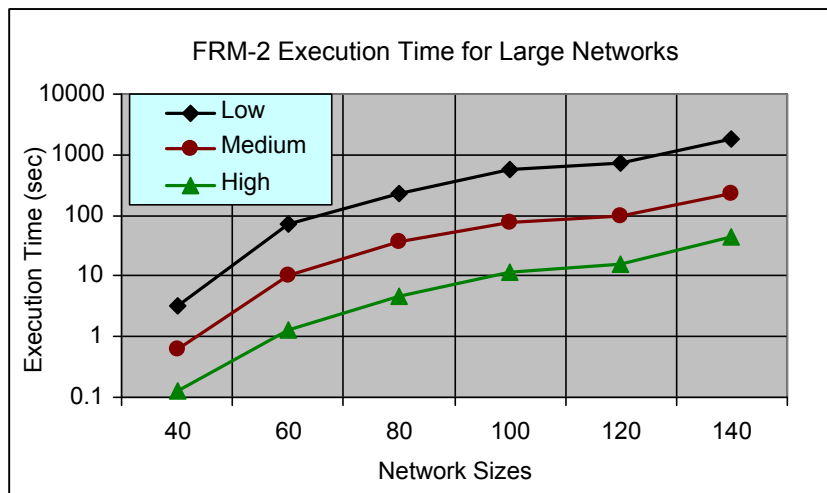


Figure 4.4: Illustrating FRM-2 execution time for large networks

Figure (4.4) shows the average execution time needed using FRM-2 for larged sized networks with different network densities.



## Chapter 5

# Optimal Lightpath Allocation in Translucent Networks

### 5.1 Introduction

In a translucent optical network (reviewed in Section 2.1.14), in order to enable every node-pair to communicate with each other, a selected number of nodes need to be capable of 3R-regeneration (reviewed in Section 2.1.13) of incoming optical signals. The number and the locations of 3R-regenerator nodes are determined by solving the Regenerator Placement Problem (RPP) (Chapter 4).

Once the locations of the regenerator nodes are known, the objective of the Routing with Regenerators Problem (RRP) is to compute a path between a given source node  $S$  to a given destination node  $D$  for a data communication request from node  $S$  to node  $D$ , using as few regenerators as possible [85,87], so that a lightpath can be established from  $S$  to  $D$ .

Since the regeneration devices are scarce and the process results in in-

creased delays, using fewest possible regenerators is an important objective in translucent lightpath allocation in translucent networks. Since O-E-O conversion takes place at regenerators, wavelength conversion is available for free at the regenerators. We assume that all-optical wavelength converters are not available, so that the wavelength continuity constraint [3] must be satisfied for each *segment*. As discussed in Section 2.1.14.1 the transparent sections of a translucent lightpath are often called segments.

In this chapter we have presented three formulations in our investigations on solving RWA problem in translucent optical networks. The first formulation (we call it DLA for Dynamic Lightpath Allocation) is to solve the problem of dynamic lightpath allocation. The second formulation (we call it DSLA for Dynamic Survivable Lightpath Allocation) augments the first formulation by adding shared path protection [88]. The third one (we call it SLA for Static Lightpath Allocation) is to solve the problem of static lightpath allocation in translucent optical networks.

An example of a long haul network with distances between the nodes in kilometers is shown in Figure 5.1. If the optical reach is  $r = 2000$  km, an optical signal from node  $A$  cannot reach node  $H$  without regeneration. For communication between  $A$  and  $H$ , if there is a regenerator at  $D$ , a translucent lightpath ( $P = A \rightarrow B \rightarrow C \rightarrow D \rightarrow F \rightarrow G \rightarrow B \rightarrow C \rightarrow H$ ) with two segments ( $S_1 = A \rightarrow B \rightarrow C \rightarrow D$ ) and ( $S_2 = D \rightarrow F \rightarrow G \rightarrow B \rightarrow C \rightarrow H$ ) can be established.

**Property 1:** *If a translucent lightpath involves two segments,  $S_a$  and  $S_b$  that have one or more common fiber(s), the same wavelength cannot be used for both segments  $S_a$  and  $S_b$  [67].*

For instance, in Figure 5.1, the segments  $S_1$  and  $S_2$  of the translucent

lightpath  $P$  have the common fiber  $B \rightarrow C$ . If a channel  $q \in Q$  is available on all the fibers in the network, when processing a request for communication from  $A$  to  $H$ , the channel  $q$  cannot be used for both segments  $S_1$  and  $S_2$ .

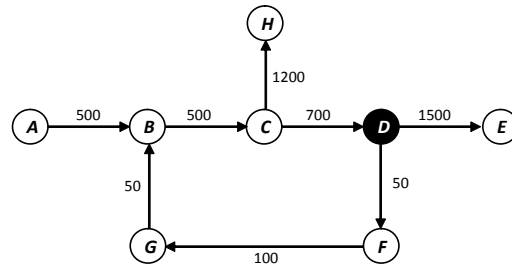


Figure 5.1: A simple translucent network with fiber lengths in km

This property of translucent networks was first identified in [67]. To the best of our knowledge, this property has not been taken into account in any earlier algorithms for translucent network design. As a result, earlier algorithms for routing in translucent networks may give an invalid route and wavelength if segments have common fiber(s). We have incorporated this important property of translucent optical network and showed how this restriction may be taken into account while solving the RRP problem.

In Section 5.2 we have presented our formulation DLA for solving RRP for dynamic lightpath allocation in translucent networks. In Section 5.3 we have presented our formulation DSLA for solving RRP for dynamic survivable lightpath allocation in translucent networks. We have presented our formulation SLA for solving the problem of static lightpath allocation in translucent networks in Section 5.4.

## 5.2 RRP for Dynamic Lightpath Allocation

In this section we describe our formulation DLA for dynamic lightpath allocation. We have shown how Property 1 may be taken into account when solving the RRP problem. We have studied the problem of dynamic lightpath allocation to establish a lightpath from some node  $S$  to some other node  $D$  in the presence of other lightpaths (translucent or transparent), set up earlier in response to previous requests for communication. We have taken account of Property 1 in

- proposing an Integer Linear Program (ILP) which gives an optimal path for a translucent lightpath using a minimum number of regenerators,
- proposing an efficient heuristic to compute the path with a minimum number of regenerators for large networks.

### 5.2.1 DLA: An ILP Formulation for Optimal Solution

This ILP formulation determines a route with the minimum number of regenerators needed to set up from a given source to given destination for a new request of data communication. In this formulation, it is necessary to specify an upper limit on the value of  $s$ , the maximum allowable number of segments in a translucent lightpath. This may be determined by the number of regenerators in the network or by the acceptable limits on the delay and the Bit Error Rates (BER) [87].

In this formulation, the symbol  $r$  denotes the optical reach of the network.  $N$  is the set of all end nodes in the network. Here  $\delta_i$  is a constant, related to a node  $i \in N$ , defined as follows. If node  $i$  is a regenerator node,  $\delta_i = 1$ , otherwise  $\delta_i = 0$ . We have used a constant  $s$  to denote the maximum allowable

number of segments in a translucent lightpath.  $E$  is the set of all pairs  $(i, j)$  of nodes such that  $i \rightarrow j$  is an edge in the physical topology, representing an optical fiber from node  $i$  to node  $j$ . The length of edge  $i \rightarrow j$  is the constant  $d_{ij}$ .  $Q$  is the set of available channels in each fiber. We have used  $w_{ij}^q$ , a constant for channel  $q \in Q$  on the fiber  $i \rightarrow j$ . If channel  $q$  is already used by an existing lightpath in edge  $i \rightarrow j$ ,  $w_{ij}^q = 1$ , otherwise  $w_{ij}^q = 0$ .

$X_{ij}^k$  denotes a binary variable for each edge  $i \rightarrow j$  in the physical topology and for each segment  $k$ ,  $1 \leq k \leq s$ , such that, if edge  $i \rightarrow j$  is used by the new lightpath in its  $k^{\text{th}}$  segment,  $X_{ij}^k = 1$ , otherwise  $X_{ij}^k = 0$ .  $W_q^k$  denotes another binary variable for each channel  $q$  in a segment  $k$  such that, if channel  $q$  is allotted to the new lightpath in its  $k^{\text{th}}$  segment,  $W_q^k = 1$ , otherwise  $W_q^k = 0$ .

The formulation is given as:

$$\text{minimize } \sum_{k=1}^s \sum_{(i,j) \in E} \delta_i \cdot X_{ij}^k \quad (5.1)$$

subject to:

1. Satisfy the flow balance equations.

$$\sum_{j:(S,j) \in E} X_{Sj}^1 = 1; \quad \sum_{k=1}^s \sum_{j:(j,S) \in E} X_{jS}^k = 0; \quad (5.2)$$

$$\sum_{k=1}^s \sum_{j:(j,D) \in E} X_{jD}^k = 1; \quad \sum_{k=1}^s \sum_{j:(D,j) \in E} X_{Dj}^k = 0; \quad (5.3)$$

$$\sum_{j:(i,j) \in E} X_{ij}^k - \sum_{j:(j,i) \in E} X_{ji}^k = 0; \quad \forall i \in N : \delta_i = 0, \quad (5.4)$$

$$\forall k, 1 \leq k \leq s.$$

$$\sum_{j:(i,j) \in E} X_{ij}^{k+1} - \sum_{j:(j,i) \in E} X_{ji}^k = 0; \quad \forall i \in N : \delta_i = 1, \quad (5.5)$$

$$\forall k, 1 \leq k \leq s.$$

2. The length of a segment cannot exceed the optical reach  $r$ .

$$\sum_{j:(i,j) \in E} X_{ij}^k \cdot d_{ij} \leq r; \quad \forall k, 1 \leq k \leq s. \quad (5.6)$$

3. Each segment of the translucent lightpath must have exactly one channel number assigned to it.

$$\sum_{q \in Q} W_q^k = 1; \quad \forall k, 1 \leq k \leq s. \quad (5.7)$$

4. The channel number assigned to a segment of the new lightpath must be free (not being used by any existing lightpath), on each fiber in the segment.

$$w_{ij}^q \cdot X_{ij}^k + W_q^k \leq 1; \quad \forall (i, j) \in E, \quad (5.8)$$

$$\forall k, 1 \leq k \leq s,$$

$$\forall q \in Q.$$

5. If two segments share a fiber, they must be assigned distinct channel numbers.

$$X_{ij}^k + X_{ij}^l + W_q^k + W_q^l \leq 3; \quad \forall (i, j) \in E, \quad (5.9)$$

$$\forall k, l, 1 \leq k, l \leq s,$$

$$\forall q \in Q.$$

### 5.2.1.1 Justification for the Formulation DLA

The objective of our formulation is to minimize the overall number of regenerator nodes used by the new translucent lightpath from source  $S$  to destination  $D$ . Equation (5.1) ensures our objective by minimizing number of edges where the starting node is a regenerator node, for all the segments needed for the lightpath.

The flow balance equations for every segment of the lightpaths are stated in the equations (5.2) through (5.5). In equation (5.4),  $\delta_i = 0$  means that the node  $i$  does not have any capability for 3R regeneration. In equation (5.5),  $\delta_i = 1$  means that the node  $i$  is a 3R regenerator node and, when a lightpath passes through such a regenerator node, the segment number starting from this regenerator node increases by 1.

Equation (5.6) ensures that no segment of the new translucent lightpath has a length exceeding the optical reach  $r$ . The purpose of equation (5.7) is to ensure that each segment of the lightpath is assigned exactly one wavelength channel. Equation (5.8) ensures that a new lightpath can only use an unused channel in any segment. In equation (5.8), if  $w_{ij}^q$  is 1, an existing lightpath is using channel  $q \in Q$  in the edge  $i \rightarrow j$ . If the new lightpath uses the same edge in any segment, then it cannot use the channel  $q$  when traversing through that segment.

As we have specified in Property 1, a translucent lightpath is permitted to have cycles in its physical routing with the restriction that whenever two segments of a lightpath share a fiber, they must be assigned distinct channel numbers. Equation (5.9) enforces this restriction as follows:

If both  $X_{ij}^k = 1$  and  $X_{ij}^l = 1$ , it means that the new lightpath is using edge

$i \rightarrow j$  in two of its segments  $k$  and  $l$ . In that case if  $W_q^k$  is 1 that is, if the lightpath uses channel  $q$  in segment  $k$ , then  $W_q^l$  must be 0, that is the lightpath must not use the same channel  $q$  in segment  $l$ , or vice versa.

### 5.2.2 A Heuristic for Dynamic Lightpath Allocation

The formulation DLA that we have presented in Section 5.2.1 gives us an optimal solution for dynamic lightpath allocation scenario for translucent networks. DLA works reasonably fast for networks having a small number of end-nodes. For a sufficiently large network, DLA may takes hours to allocate a single lightpath, which is unacceptable in a dynamic environment, where a request must be served within a few milliseconds.

In this section we propose a fast heuristic that can handle large-sized networks and that can produce near optimal solution with a fraction of the time needed for an optimal solution.

In this heuristic, we have used the term *path intersection graph* to denote a graph  $G_{\mathbb{P}} = (\mathbb{V}_{\mathbb{P}}, \mathbb{E}_{\mathbb{P}})$ , where each vertex in  $\mathbb{V}_{\mathbb{P}}$  represents a path through the optical network that may be used to set up a transparent lightpath. If  $p$  and  $q$  are two vertices in  $\mathbb{V}_{\mathbb{P}}$ , there will be an edge, in  $\mathbb{E}_{\mathbb{P}}$ , between  $p$  and  $q$ , iff the paths, corresponding to  $p$  and  $q$ , share one or more physical edge(s). We have established, through extensive simulations with dynamic call arrivals, the effectiveness of the heuristic by measuring the call blocking probabilities. We have also demonstrated the relative impact of different choices of network resources, such as (i) the number of regenerators, (ii) the optical reach of the regenerators and (iii) the number of wavelengths, on the network performance, measured in terms of the call blocking probability.

In our heuristic, we have used the following symbols in addition to the



symbols used for the ILP formulation in Section 5.2.1. Here  $\mathbb{P}_{xy}$  denotes the set of pre-computed paths from node  $x$  to node  $y$ ,  $\mathbb{S}$  denotes a set of states that need to be explored<sup>1</sup>,  $\mathcal{N}_{\mathcal{S}}$  a newly created state,  $\mathcal{C}$  is a set of outgoing paths from the current node.  $\overline{Q}_{ij}$  represents the set of channels still available on edge  $i \rightarrow j$ .  $\mathcal{W}_v$  is the set of colors to color vertex  $v$  of  $\mathbb{G}_{\mathbb{P}}$ .  $\mathcal{W}$  is the set of the set of colors  $\{\mathcal{W}_v : v \in \mathbb{V}_{\mathbb{P}}\}$ .

The objective of the heuristic outlined below is to establish, if possible, a translucent lightpath, using a minimum number of regenerators. The heuristic assumes that necessary regenerators have been already deployed in the network. This heuristic has used a simple scheme of considering a fixed number of routes when establishing a transparent lightpath [3,91]. Our primary objective was to show how to handle the problem of overlapping segments discussed in Section 5.2. This heuristic is based on the “central agent” approach [3] where an end-node of the network is designated as the site where the heuristic will be executed. A request for a communication from  $S$  to  $D$  has to be communicated from  $S$  to the site where the heuristic will be executed and, if the heuristic succeeds in establishing a route for the translucent lightpath, messages have to be communicated to the routers and the 3R regenerators in the path, followed by a time lag sufficient to set up the routers and the 3R regenerators before communication can start from  $S$ .

The performance of the heuristic below is based on global information and therefore gives a lower bound on the performance which is useful for calibrating any distributed heuristic based on a similar approach. The site where the

---

<sup>1</sup>Each state in the state-space for the problem consists of the triple  $(x, \mathbb{P}, \mathbb{G}_{\mathbb{P}})$  where  $x$  is either the source node  $S$  or a regenerator node in the network currently being explored,  $\mathbb{P}$  is the set of transparent paths, each with a total length  $\leq r$ , used by the search to reach node  $x$  starting from the source node  $S$  and  $\mathbb{G}_{\mathbb{P}}$  is the path intersection graph corresponding to  $\mathbb{P}$ .

heuristic is executed has access to a database containing:

1. the set  $\mathbb{P}_{xy}$  of  $m$  (or fewer, if all  $m$  paths cannot be found) paths, for all regenerator node pairs  $(x, y)$  where the total length of each path  $\leq r$ .
2. the set of channels,  $\overline{Q}_{pq}$ , currently available on each fiber  $p \rightarrow q$  in the network.
3. the distance  $d_{ij}$  between all pairs of end-nodes.
4. a list of all nodes capable of 3R regeneration.

Each path in  $\mathbb{P}_{xy}$  is a potential candidate for setting up a transparent lightpath from  $x$  to  $y$ , where  $x$  could be the source node or a regenerator node in the network and  $y$  could be a regenerator node or the destination node. Given a path of length  $\leq r$ , say  $x \rightarrow u \rightarrow \dots \rightarrow v \rightarrow y$ , the values of  $\overline{Q}_{xu}, \dots, \overline{Q}_{vy}$  allow us to determine the set of channel numbers that may be used to set up a new transparent lightpath from  $x$  to  $y$ .

The heuristic uses  $A^*$ , a well-known best-first search [61]. In the heuristic, each state in the state-space for the problem consists of the triple  $(x, \mathbb{P}, \mathbb{G}_P)$  where  $x$  is a node in the network,  $\mathbb{P}$  is the set of transparent paths used by the search to reach node  $x$  starting from node  $S$  and  $\mathbb{G}_P$  is the path intersection graph corresponding to  $\mathbb{P}$ . Here the cost of a translucent lightpath is the number of regenerators needed in the path used by the lightpath. Given a state  $(x, \mathbb{P}, \mathbb{G}_P)$ , the cost to reach node  $x$  from node  $S$  is  $|\mathbb{P}| - 1$ . The heuristic estimate of the cost (number of regenerations needed) to reach  $D$  from the regenerator node  $x$ . If the shortest path<sup>2</sup> from  $x$  to  $D$  involves the edges

<sup>2</sup>We have calculated the shortest path between a node-pair  $(x, y)$  using Dijkstra's shortest path algorithm [22]

$x \rightarrow u \rightarrow v \rightarrow \dots \rightarrow w \rightarrow z \rightarrow D$ , then the shortest distance from  $x$  to  $D$  can be given as  $d_{xD} = d_{xu} + d_{uv} + \dots + d_{wz} + d_{zD}$ . Since  $d_{xD}$  is the shortest path from  $x$  to  $D$ , the minimum number of regenerator needed to set up a translucent lightpath from  $x$  to  $D$  must be  $\lceil d_{xD}/r \rceil$ . The actual number of regenerators will, of course, by the path ultimately selected from  $x$  to  $D$  and the length of that path can not be than  $d_{xD}$ . This establishes that  $\lceil d_{xD}/r \rceil$  can not exceed the actual cost of regenerators from  $x$  to  $D$  and hence this is an admissible heuristic.

A state  $(x, \mathbb{P}, \mathbb{G}_P)$  is valid if it may be used to set up a translucent lightpath from  $S$  to  $x$ . Let a vertex  $v \in \mathbb{G}_P$  correspond to the path  $p = u \rightarrow v \rightarrow \dots \rightarrow y \rightarrow z$ . The set of channel numbers that may be used to set up a transparent lightpath from  $u$  to  $z$ , using path  $p$ , may be viewed as the set of colors  $\mathcal{W}_v$  to color vertex  $v$  of the path intersection graph  $\mathbb{G}_P = (\mathbb{V}_P, \mathbb{E}_P)$ . If two nodes  $u, v \in \mathbb{V}_P$  are adjacent in  $\mathbb{G}_P$ , it means that the paths corresponding to  $u$  and  $v$  share one or more fiber (s). In this situation, to satisfy the property given in Section 1, transparent lightpaths using the paths through the optical network, corresponding to  $u$  and  $v$ , cannot be assigned the same channel number. We have used a list coloring algorithm [39], with  $\mathcal{W}_v$  as the list of colors for vertex  $v$  to color graph  $\mathbb{G}_P$ . If the coloring algorithm succeeds, every path in  $\mathbb{P}$  may be used to set up a transparent lightpath, using the channel number obtained using the coloring algorithm, and the state  $(x, \mathbb{P}, \mathbb{G}_P)$  is valid. Only essential points of the heuristic are included in Algorithm 4.

In Algorithm 5, we have described the function  $createNewStates(\mathcal{C}, \mathbb{P}, \mathbb{G}_P)$ . The remaining functions in Algorithms 4 and 5 are informally described below.

$removeBest(\mathbb{S})$ : This function takes a set of states  $\mathbb{S}$  and returns the state  $X = (x, \mathbb{P}, \mathbb{G}_P), X \in \mathbb{S}$  with the lowest estimated value of the number of

---

**Algorithm 4** *Dynamic lightpath allocation from S to D*

---

```

1:  $\mathbb{S} \leftarrow (S, \{\}, (\{\}, \{\}))$ 
2: while  $\mathbb{S} \neq \{\}$  do
3:    $(x, \mathbb{P}, \mathbb{G}_P) \leftarrow \text{removeBest}(\mathbb{S})$ 
4:   if  $x = D$  then
5:     return  $(\mathbb{P}, \mathbb{G}_P)$ 
6:   else
7:     if  $d_{xD} \leq r$  then
8:        $\mathcal{C} \leftarrow \mathbb{P}_{xD} \cup \text{pathsToRegenerators}(x)$ 
9:     else
10:       $\mathcal{C} \leftarrow \text{pathsToRegenerators}(x)$ 
11:    end if
12:     $\mathcal{N}_S \leftarrow \text{createNewStates}(\mathcal{C}, \mathbb{P}, \mathbb{G}_P)$ 
13:     $\mathbb{S} \leftarrow \mathbb{S} \cup \mathcal{N}_S$ 
14:  end if
15: end while

```

---



---

**Algorithm 5** *createNewStates( $\mathcal{C}, \mathbb{P}, \mathbb{G}_P$ )*

---

```

1:  $\mathcal{N}_S \leftarrow \{\}$ 
2: for each  $\text{path } p \in \mathcal{C}$  do
3:    $\mathbb{G}_P^{\text{new}} \leftarrow \text{augmentGraphByPath}(p, \mathbb{G}_P)$ 
4:    $\mathbb{P}^{\text{new}} \leftarrow \mathbb{P} \cup \{p\}$ 
5:   for each node  $v \in \mathbb{V}_{\mathbb{P}^{\text{new}}}$  do
6:      $\mathcal{W}_v \leftarrow \text{assignListColors}(v)$ 
7:   end for
8:   if  $\text{listColor}(\mathbb{G}_P^{\text{new}}, \mathcal{W})$  then
9:      $\mathcal{N}_S \leftarrow \mathcal{N}_S \cup (\text{lastNode}(p), \mathbb{P}^{\text{new}}, \mathbb{G}_P^{\text{new}})$ 
10:  end if
11: end for
12: return  $\mathcal{N}_S$ 

```

---

regenerators needed to reach node  $D$  from node  $S$ . The function removes the state  $X$  from  $\mathbb{S}$  as well.

*pathsToRegenerators( $x$ )*: This function takes a node  $x$  of the network and returns the set of pre-computed paths from  $x$  to all regenerators that are within the optical reach of  $x$ .

*augmentGraphByPath( $p, \mathbb{G}_P$ )*: This function takes a path  $p$  in the network from some node  $x$  and the path intersection graph  $\mathbb{G}_P = (\mathbb{V}_P, \mathbb{E}_P)$  corresponding to the paths used to go from  $S$  to  $x$ . The function returns a graph  $\mathbb{G}_P^{new}$  by adding, to  $\mathbb{G}_P$ , a new vertex  $v$  corresponding to path  $p$  and requisite new edges between  $v$  and the nodes in  $\mathbb{V}_P$ .

*assignListColors( $v$ )*: This function takes vertex  $v$  of the path intersection graph  $\mathbb{G}_P = (\mathbb{V}_P, \mathbb{E}_P)$  (which corresponds to some path in the network with a total length  $\leq r$ ) and returns the set of channel numbers that are not used on any of the edges in the path. This set is used to define  $\mathcal{W}_v$  that can be used to color vertex  $v$ .

*listColor( $\mathbb{G}_P, \mathcal{W}$ )*: This function takes a path intersection graph  $\mathbb{G}_P = (\mathbb{V}_P, \mathbb{E}_P)$  and, for each vertex  $v \in \mathbb{V}_P$ , a set of colors  $\mathcal{W}_v$ . The function returns true, if it can successfully use a list coloring algorithm to assign a color from  $\mathcal{W}_v$  to vertex  $v, \forall v \in \mathbb{V}_P$ .

*lastNode( $p$ )*: This function takes a transparent path  $p$  through the optical network and returns the last node in  $p$ .

### 5.2.3 Experimental Results

We conducted two sets of experiments<sup>3</sup> to study the efficacy of our heuristics on many realistic networks of varying sizes, namely, ARPANET(20 nodes, 32 links), LATA 'X'(28, 47) and USANET(53,68) [41]. The link distances in these networks were randomly chosen between 1 and 1000 km with equal probability. In the first set of experiments, the design parameters we considered were (i) *the number of pre-computed paths between regenerators* (ii) *the traffic load on the network* (iii) *the number of regenerators in the network* and (iv) *optical reach distance*. We studied the effect of these design parameters on the *call blocking probability* for a dynamic call scenario. We dynamically generated 10,000 calls. Each call  $i$  was a tuple  $(s_i, t_i, a_i, d_i)$  where  $s_i, t_i$  are the source and destination of the call respectively,  $a_i$  is the arrival time and  $d_i$  is the duration of the call. We assumed that the call arrival process followed the Poisson distribution and the call durations had exponential holding time. We define the traffic load in the network to be the multiple of the call arrival rate and the average call holding time. We varied the traffic load by changing the arrival rate and the average call holding time. In all these experiments, we selected the nodes with regeneration capabilities randomly with uniform probability.

Figure 5.2 shows the effect of the number of pre-computed paths maintained by the heuristic on the call blocking probability. In this experiment, we assumed that the number of wavelengths available on each link was 8 and the traffic load to be 50 erlangs<sup>4</sup>. For the different networks, Figure 5.2 shows

---

<sup>3</sup>The actual experiments were conducted by the other group at Arizona State University. My contribution was to develop the algorithms, ILP formulations DLA and the heuristic [4].

<sup>4</sup>Erlang is a dimensionless unit of network traffic, named after the Danish telephone engineer A. K. Erlang, and is given by the equation  $E = \lambda h$ , where  $E$  is the erlang,  $\lambda$  represents the call (request) arrival rate and  $h$  is the average call holding time [25].

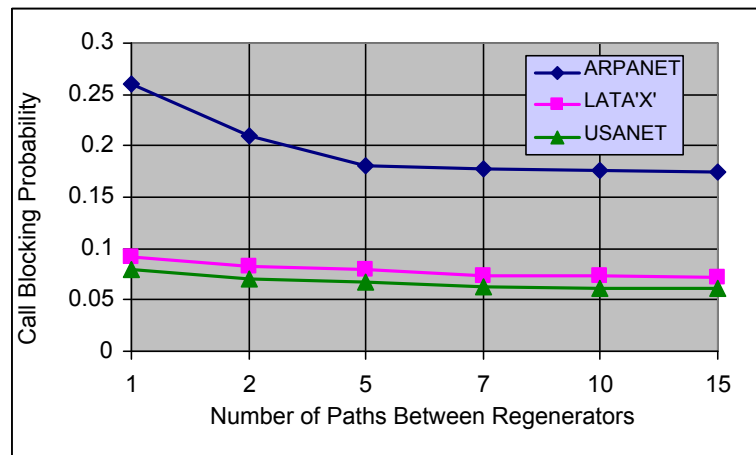


Figure 5.2: Effect of varying the number of pre-computed paths considered

that the maximum improvement, in terms of the call blocking probability can be achieved by maintaining 5 pre-computed paths (if exists) between every pair of regenerators. The improvement in the call blocking probability is not significant beyond 5 pre-computed paths. Correspondingly, in the subsequent experiments, we selected the upper limit on the number of pre-computed paths to be 5. In other words, if there are 5 or more paths for a source  $S$  to a destination  $D$ , we selected the 5 shortest paths from the source to the destination. It is quite possible that we do not have 5 paths from a source  $S$  to a destination  $D$ . In that case we selected all the paths from  $S$  to  $D$ . For all the networks, the heuristic completed within 5 seconds, when number of paths was 5 or less and took 2 minutes or less when the number of paths was more than 5.

Figure 5.3 and Figure 5.4 show the effect of varying traffic load on the call blocking probability for different values of the number of wavelengths per link in the LATA 'X' and in the ARPANET networks respectively. Here the traffic loads were varied from 0 to 50 erlangs in steps of 5 erlangs. The call blocking probabilities were measured for 4, 8, 12 and 16 wavelengths per link in the

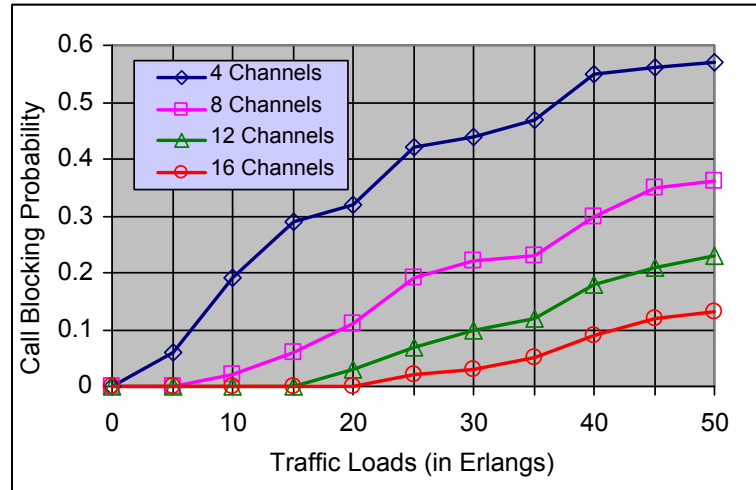


Figure 5.3: Traffic load vs. call blocking probability for LATA 'X'

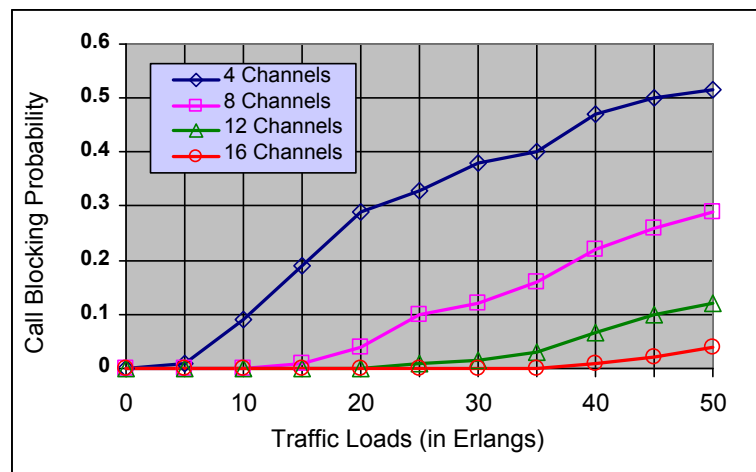


Figure 5.4: Traffic load vs. call blocking probability for ARPANET



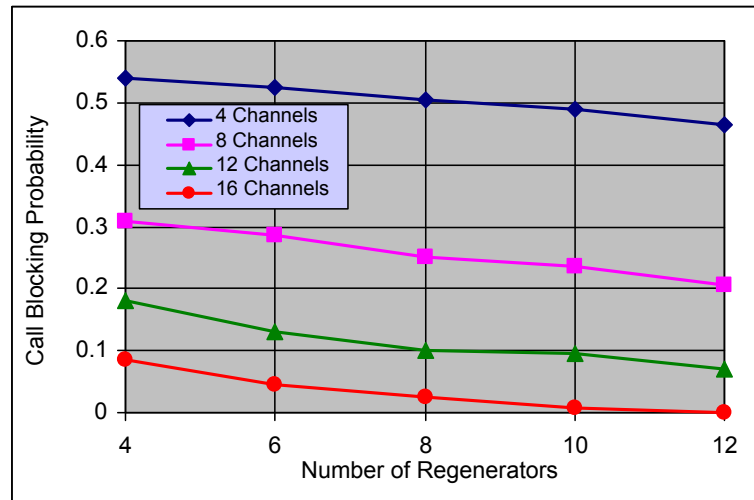


Figure 5.5: Number of regenerator vs. call blocking probability for ARPANET

networks. The value of  $r$  was set to be 1000 km and approximately 33% of nodes were randomly selected as regenerators. As expected, at higher loads, increasing the number of wavelengths significantly reduces the call blocking probability.

Figure 5.5 shows the effect of the number of regenerator nodes in the ARPANET network on the call blocking probability for traffic load = 50 erlangs and optical reach distance = 1000 km. It is interesting to observe that increasing the number of wavelengths per link in the network has a higher impact on the call blocking probability than increasing the number of regenerators in the network. For instance, doubling the number of regenerators from 6 to 12 marginally reduces the call blocking probability from 0.515 to 0.458, whereas doubling the number of wavelengths from 4 to 8 reduces the call blocking probability from 0.515 to 0.286 in ARPANET.

Figure 5.6 shows the effect of varying the value of the optical reach on the call blocking probability for the ARPANET network with traffic load = 50

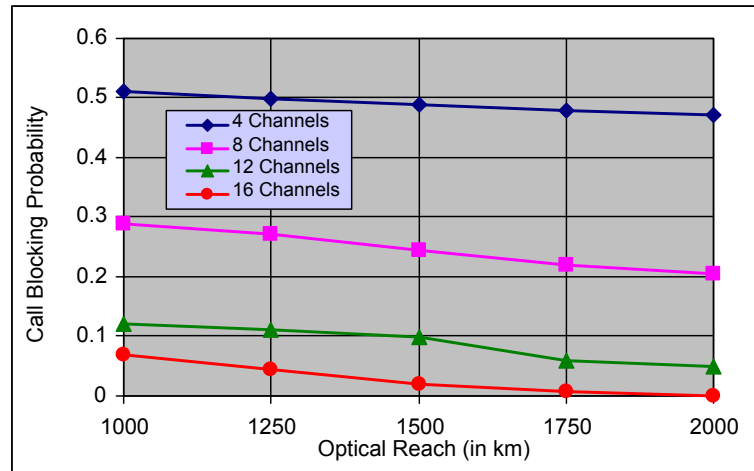


Figure 5.6: Optical reach vs. blocking probability for ARPANET

erlangs and 6 regenerators. From the figures, we can observe that increasing the optical reach, while keeping the number of regenerator node same, has marginal benefit in terms of reducing the call blocking probability. These tradeoffs between the different design parameters can be effectively used by a network designer to select the right set of additional resources to improve the network performance.

In the second set of experiments, our goal was to compare the number of regenerators on the path produced by the heuristic with that of the optimal solution found by solving DLA. The experiments were conducted on 3 different networks - COST-SMALL<sup>5</sup>, ARPANET and LATA ‘X’. The ILP was solved using CPLEX-10 optimizer [36].

The results of these experiments are shown in Figure 5.7. The numbers over the bars indicate the total execution time in seconds taken by the heuristic and ILP to compute the end-to-end paths for all node pairs. When computing

<sup>5</sup>COST-SMALL network consists of 11 nodes and 22 edges. USANET results are omitted since the ILP did not produce optimal solution even after a significant amount of time.

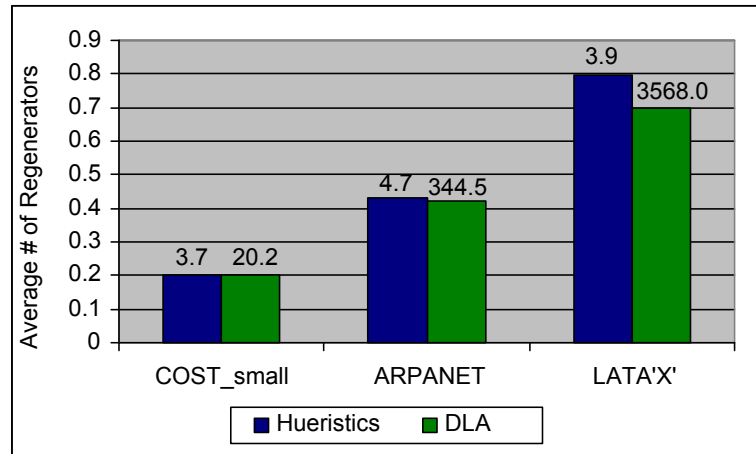


Figure 5.7: Average number of regenerators computed by the heuristic and DLA for different networks

the average number of regenerators, only those node pairs were considered for which the heuristics found a solution. It is evident from the results that the heuristics produce near-optimal solutions in a fraction of the time needed to find the optimal solutions, even for medium-size networks. Also, for the LATA 'X' network, the heuristic failed to find a path, even when such a path existed (as shown in the optimal results) for a total 11 source-destination pairs out of 338 source-destination paths.

### 5.3 RRP for Dynamic Survivable Lightpath Allocation

Schemes to handle faults in optical networks [35, 74, 79, 96] have received a lot of attention. Any lightpath allocation scheme for optical networks that implements any system of protecting a lightpath from a fault in the network is known as survivable lightpath allocation scheme. *Shared path protection* [88] is a popular scheme due to its efficient use of resources and relatively fast recovery time. In dynamic lightpath allocation using shared path protection,

in response to a request for communication, provisions have to be made for two lightpaths - a *primary* lightpath and a *backup* lightpath which are fiber-disjoint. If two primary paths are edge-disjoint, the corresponding backup paths are allowed to share one or more fiber(s) as well as the channel number. Our objective is to use a minimum number of regenerators in the new primary and backup lightpaths. We consider two Integer Linear Program (ILP) formulations to address the Routing with Regenerator Problem, with shared path protection, taking Property 1 (as describe in section 5.2) into consideration. The first formulation (we call it DSLA-1 for Dynamic Survivable Lightpath Allocation formulation 1) gives us optimal solutions but it might be time consuming while the second formulation (we call it DSLA-2 for Dynamic Survivable Lightpath Allocation formulation 2) produces near-optimal solutions very fast.

In both formulations DSLA-1 and DSLA-2, we assume that a number of requests for communication have been already processed. If a translucent lightpath enters a node with 3R regenerator capability, it does not necessarily mean that the lightpath has to use 3R regeneration at that node. Each successful request for communication results in i) a route and wavelength assignment for a primary and a backup translucent lightpath, ii) the deployment of the primary lightpath and iii) depending on the network policy, either the reservation of the resources for, or the deployment of, the backup lightpath, to be used if there is a fault affecting the primary lightpath. When a communication is over, all resources for the corresponding primary and the backup lightpath will be released for handling future requests for communication. There is a database containing, for each ongoing communication, information about the primary lightpath and the resources reserved for the backup lightpath. This

database is used by both formulations DSLA-1 and DSLA-2 when processing a new request for a communication, say, from  $S$  to  $D$ . The objective of both DSLA-1 and DSLA-2 is to determine, if possible, the route of the primary (backup) translucent lightpath from  $S$  to  $D$ , and the channel number allocated to each segment of the primary (backup) lightpath. The total number of 3R regenerators used by the primary and the backup translucent lightpath should be as small as possible.

Formulation DSLA-1 given below considers all possible paths from  $S$  to  $D$  and will succeed if it is possible to set up a primary and a backup translucent lightpath from  $S$  to  $D$ . For fast processing, in formulation DSLA-2, the search space has been restricted to some extent, so that it may occasionally fail, even if a valid primary and a backup translucent lightpath from  $S$  to  $D$  exists.

### 5.3.1 DSLA-1: An ILP Formulation for Optimal Solution

The constants in DSLA-1 describe the network and the lightpaths that have been deployed already to support existing requests for communication. The network has a set  $N$  of nodes, with a set  $E$  of edges, with each edge  $(i, j) \in E$  representing a fiber, capable of carrying  $|Q|$  channels.  $M$  denotes a large positive number. The maximum possible number of segments in the translucent lightpath from  $S$  to  $D$  will be a specified constant  $s$ , determined by the upper limit of acceptable Bit Error Rate (BER) and the communication delay. The distance of edge  $(i, j) \in E$  will be  $d_{ij}$ .  $\mathcal{L}$  denotes the total number of communications currently in progress. The channel number allocated to the  $k^{\text{th}}$  segment of the  $l^{\text{th}}$  existing backup lightpath ( $1 \leq l \leq \mathcal{L}$ ) will be  $\omega_k^l$ . Here  $\delta_i$ ,  $w_{ij}^q$ ,  $z_{ij}^q$ ,  $a_{ij}^l$ ,  $b_{ij}^{lk}$  denote constants as follows. If node  $i$  is a 3R regenerator,  $\delta_i = 1$ ; otherwise  $\delta_i = 0$ . If an existing primary (backup) lightpath uses chan-

nel  $q$  on edge  $(i, j)$ ,  $w_{ij}^q(z_{ij}^q) = 1$ ; otherwise  $w_{ij}^q(z_{ij}^q) = 0$ . If the  $l^{\text{th}}$  existing primary lightpath,  $(1 \leq l \leq \mathcal{L})$ , uses edge  $(i, j)$ ,  $a_{ij}^l = 1$ ; otherwise  $a_{ij}^l = 0$ . If segment  $k$  of the  $l^{\text{th}}$  backup lightpath,  $(1 \leq l \leq \mathcal{L})$ , uses edge  $(i, j)$ ,  $b_{ij}^{lk} = 1$ ; otherwise  $b_{ij}^{lk} = 0$ .

$W_q^k$ ,  $Z_q^k$ ,  $X_{ij}^k$ ,  $Y_{ij}^k$  are binary variables. If segment  $k$  of the new primary (backup) lightpath uses channel  $q$ ,  $W_q^k(Z_q^k) = 1$ ; otherwise  $W_q^k(Z_q^k) = 0$ . If segment  $k$  of the new primary (backup) lightpath uses edge  $(i, j)$ ,  $X_{ij}^k(Y_{ij}^k) = 1$ ; otherwise  $X_{ij}^k(Y_{ij}^k) = 0$ .  $\gamma_{lh}^k$  denotes a non-negative continuous variable for the new backup path, whose values are restricted by the constraints, such that  $\gamma_{lh}^k = 1$ , if segment  $k$  of the new backup path shares an edge and a channel number used by segment  $h$  of the  $l^{\text{th}}$  existing backup path; otherwise  $\gamma_{lh}^k = 0$ .

The objective of DSLA-1 is to minimize a composite function involving the total number of regenerators in the new primary and the new backup lightpaths, and the total number of channels in the new primary and the new backup lightpaths. By making  $M$  sufficiently large, in equation (5.10), we ensure that the first priority is to minimize the number of regenerators and the second priority is to minimize the total number of physical links used in the new primary and the new backup lightpaths.

The formulation is given as:

$$\text{minimize } M \cdot \sum_{k=1}^s \sum_{(ij) \in E} (\delta_i \cdot X_{ij}^k + \delta_i \cdot Y_{ij}^k) + \sum_{k=1}^s (X_{ij}^k + Y_{ij}^k) \quad (5.10)$$

subject to:

1. Satisfy the flow balance equations for both the new primary lightpath and the new backup lightpath and ensure that the segment number following a regenerator node is 1 more than the segment number preceding the

regenerator node.

$$\sum_{j:(S,j) \in E} X_{Sj}^1 = 1; \quad \sum_{k=1}^s \sum_{j:(j,S) \in E} X_{jS}^k = 0; \quad (5.11)$$

$$\sum_{j:(S,j) \in E} Y_{Sj}^1 = 1; \quad \sum_{k=1}^s \sum_{j:(j,S) \in E} Y_{jS}^k = 0; \quad (5.12)$$

$$\sum_{k=1}^s \sum_{j:(D,j) \in E} X_{Dj}^k = 0; \quad \sum_{k=1}^s \sum_{j:(j,D) \in E} X_{jD}^k = 1; \quad (5.13)$$

$$\sum_{k=1}^s \sum_{j:(D,j) \in E} Y_{Dj}^k = 0; \quad \sum_{k=1}^s \sum_{j:(j,D) \in E} Y_{jD}^k = 1; \quad (5.14)$$

$$\sum_{j:(ij) \in E} X_{ij}^k - \sum_{j:(ji) \in E} X_{ji}^k = 0; \quad \forall i \in N : \delta_i = 0, \quad (5.15)$$

$$\forall k, 1 \leq k \leq s.$$

$$\sum_{j:(ij) \in E} X_{ij}^{k+1} - \sum_{j:(ji) \in E} X_{ji}^k = 0; \quad \forall i \in N : \delta_i = 1, \quad (5.16)$$

$$\forall k, 1 \leq k \leq s.$$

$$\sum_{j:(ij) \in E} Y_{ij}^k - \sum_{j:(ji) \in E} Y_{ji}^k = 0; \quad \forall i \in N : \delta_i = 0, \quad (5.17)$$

$$\forall k, 1 \leq k \leq s.$$

$$\sum_{j:(ij) \in E} Y_{ij}^{k+1} - \sum_{j:(ji) \in E} Y_{ji}^k = 0; \quad \forall i \in N : \delta_i = 1, \quad (5.18)$$

$$\forall k, 1 \leq k \leq s.$$

2. The length of any segment of the new primary lightpath or the new

backup lightpath cannot exceed the optical reach  $r$ .

$$\sum_{(ij) \in E} X_{ij}^k \cdot d_{ij} \leq r; \quad \forall k, 1 \leq k \leq s. \quad (5.19)$$

$$\sum_{(ij) \in E} Y_{ij}^k \cdot d_{ij} \leq r; \quad \forall k, 1 \leq k \leq s. \quad (5.20)$$

3. Each segment of the new primary lightpath or the new backup lightpath must have exactly one channel number assigned to it.

$$\sum_{q \in Q} W_q^k = 1; \quad \forall k, 1 \leq k \leq s. \quad (5.21)$$

$$\sum_{q \in Q} Z_q^k = 1; \quad \forall k, 1 \leq k \leq s. \quad (5.22)$$

4. The new primary lightpath must not share an edge with the new backup lightpath.

$$X_{ij}^k + Y_{ij}^h \leq 1; \quad \forall (i, j) \in E, \quad (5.23)$$

$$\forall k, h, 1 \leq k, h \leq s.$$

5. The channel number assigned to each segment of the new primary lightpath must not be in use by any existing lightpath (primary or backup), on each fiber in the segment.

$$w_{ij}^q \cdot X_{ij}^k + W_q^k \leq 1; \quad \forall (i, j) \in E, \quad (5.24)$$

$$\forall k, 1 \leq k \leq s,$$

$$\forall q \in Q.$$



$$\begin{aligned}
z_{ij}^q \cdot X_{ij}^k + W_q^k &\leq 1; \quad \forall (i, j) \in E, \\
&\forall k, 1 \leq k \leq s, \\
&\forall q \in Q.
\end{aligned} \tag{5.25}$$

6. The new backup lightpath, in any fiber in its path, can not share a channel number that is being used by any existing primary lightpath.

$$\begin{aligned}
w_{ij}^q \cdot Y_{ij}^k + Z_q^k &\leq 1; \quad \forall (i, j) \in E, \\
&\forall k, 1 \leq k \leq s, \\
&\forall q \in Q.
\end{aligned} \tag{5.26}$$

7. The new backup lightpath may share a channel number with another existing backup lightpath only if the corresponding existing primary lightpath and the new primary lightpath are edge-disjoint. Equations (5.27), (5.28) and (5.29) ensure that, for all  $k, h, 1 \leq k, h \leq s$  and for all  $l, 1 \leq l \leq \mathcal{L}$ , the continuous variable  $\gamma_{lh}^k$  has a value 1 if and only if the  $k^{th}$  segment of the new backup lightpath shares some edge  $(i, j)$  as well as the channel number with the  $h^{th}$  segment of the  $l^{th}$  existing backup lightpath; otherwise  $\gamma_{lh}^k$  has a value 0.

$$\begin{aligned}
Z_{\omega_h^l}^k + Y_{ij}^k - \gamma_{lh}^k &\leq 1; \quad \forall (i, j) \in E : b_{ij}^{lh} = 1, \\
&\forall k, h, 1 \leq k, h \leq s, \\
&\forall l, 1 \leq l \leq \mathcal{L}.
\end{aligned} \tag{5.27}$$

$$\begin{aligned} \gamma_{lh}^k - Z_{\omega_h^l}^k &\leq 0; \quad \forall k, h, 1 \leq k, h \leq s, \\ &\forall l, 1 \leq l \leq \mathcal{L}. \end{aligned} \quad (5.28)$$

$$\begin{aligned} \gamma_{lh}^k - \sum_{(ij):b_{ij}^{lh}=1} Y_{ij}^k &\leq 0; \quad \forall (i, j) \in E, \\ &\forall k, h, 1 \leq k, h \leq s, \\ &\forall l, 1 \leq l \leq \mathcal{L}. \end{aligned} \quad (5.29)$$

$$\begin{aligned} a_{ij}^l + X_{ij}^k + \gamma_{lh}^k &\leq 2; \quad \forall (i, j) \in E, \\ &\forall k, h, 1 \leq k, h \leq s, \\ &\forall l, 1 \leq l \leq \mathcal{L}. \end{aligned} \quad (5.30)$$

8. If two segments of the new lightpath (primary or backup) share a fiber, they must be assigned distinct channel numbers. As shown in Figure 5.1, a translucent lightpath may have cycles and this constraint ensures that, whenever two segments of a lightpath share a fiber, they must be assigned distinct channel numbers. Equations (5.31) and (5.32) enforce this restriction for the primary lightpath and the backup lightpath respectively.

$$\begin{aligned} X_{ij}^k + X_{ij}^h + W_q^k + W_q^h &\leq 3; \quad \forall (i, j) \in E, \\ &\forall k, h, 1 \leq k, h \leq s : k \neq h, \\ &\forall q \in Q. \end{aligned} \quad (5.31)$$

$$\begin{aligned}
Y_{ij}^k + Y_{ij}^h + Z_q^k + Z_q^h &\leq 3; \quad \forall (i, j) \in E, & (5.32) \\
\forall k, h, 1 \leq k, h \leq s : k &\neq h, \\
\forall q \in Q.
\end{aligned}$$

### 5.3.1.1 Justification for the Formulation DSLA-1

The objective of the formulation DSLA-1 is to:

1. Minimize the number of regenerators needed to set up a translucent logical edge from any source  $S$  to any destination  $D$ . Corresponding to this logical edge from  $S$  to  $D$  a primary lightpath will be deployed and resources for a backup lightpath will be reserved.
2. Minimize the number of physical links through which these lightpaths would be established.

Our objective is to minimize a composite function involving the total number of regenerators in the primary and backup lightpath, and the number of channel also in the primary and backup lightpaths. By making  $M$  sufficiently large we ensure in the equation (5.10) that the number of regenerators are minimized first and then the number of physical links are minimized.

The flow balance equations for every segment of primary and backup lightpaths are stated in the equations (5.11) through (5.18). In equations (5.15) and (5.17),  $\delta_i = 0$  means that the node  $i$  does not have any capability for 3R regeneration. In equations (5.16) and (5.18),  $\delta_i = 1$  means that the node  $i$  is a 3R regenerator node and, when a lightpath passes through such a regenerator node, its segment number increases by 1.

Equations (5.19) and (5.20) ensure that the total length of any segment of a translucent lightpath (primary or backup) cannot exceed the optical reach  $r$ . The purpose of equations (5.21) and (5.22) is to ensure that each segment of a lightpath is assigned exactly one wavelength channel. Equation (5.23) ensures that the new primary path must be edge-disjoint with respect to its backup path. If the primary path uses an edge  $i \rightarrow j$  in its physical route then the corresponding backup path is restricted from using the same edge. Equations (5.24) and (5.25) ensure that a new primary lightpath can only use an unused channel in any segment. In equation (5.24), if  $w_{ij}^q$  is 1, an existing primary path is using a channel  $q$  in the edge  $i \rightarrow j$ . If the new primary path uses the same edge in any segment, then it cannot use the channel  $q$  in that segment. Equation (5.25) is for a similar situation with an existing backup path. Similarly, equation (5.26) ensures that the new backup path cannot share a channel in any segment with an existing primary path.

Equations (5.27) through (5.30) need some explanations. If, in segment  $k$ , the new backup lightpath uses a channel number that has been already used by segment  $h$  of the existing  $l^{th}$  backup path,  $Z_{\omega_h^l}^k = 1$ . Then (5.28) indicates that  $\gamma_{lh}^k \leq 1$ . Now, if an edge  $(i, j)$ , that has been used by segment  $h$  of the  $l^{th}$  existing backup lightpath (i.e.,  $b_{ij}^{lh} = 1$ ), is also shared by the new backup lightpath in some segment  $k$ , then  $Y_{ij}^k = 1$ . The purpose of (5.27) is to state that, in this situation,  $\gamma_{lh}^k \geq 1$ , so that the only value of  $\gamma_{lh}^k$  that satisfies both (5.27) and (5.28) is  $\gamma_{lh}^k = 1$ .

If segment  $k$  of the new backup lightpath does not share any edge used by segment  $h$  of the existing  $l^{th}$  backup lightpath, equation (5.29) states that  $\gamma_{lh}^k \leq 0$ . Since all variables must be greater than or equal to 0,  $\gamma_{lh}^k$  must be 0. Now if the same segment  $k$  of the new backup lightpath also does not share the

same channel number as used by segment  $h$  of the  $l^{th}$  backup path,  $Z_{\omega_h^l}^k = 0$ . In this situation, (5.27) states that  $\gamma_{ih}^k \geq 0$ . Since (5.29) states that  $\gamma_{ih}^k \leq 0$ , the only solution that satisfies both (5.27) and (5.29), in this situation, is  $\gamma_{ih}^k = 0$ .

In summary, the equations (5.27), (5.28) and (5.29) ensure that, for all  $k$ ,  $1 \leq k \leq s$  and for all  $l$   $1 \leq l \leq \mathcal{L}$ , the continuous variable  $\gamma_{ih}^k$  has a value 1 if and only if

- the new backup lightpath in its  $k^{th}$  segment, shares some edge  $i \rightarrow j$  in the physical topology with the  $h^{th}$  segment of the  $l^{th}$  existing backup lightpath and
- the new backup lightpath in the same  $k^{th}$  segment shares a channel number with the  $h^{th}$  segment of the  $l^{th}$  existing backup lightpath.

Otherwise  $\gamma_{ih}^k$  has a value 0.

If the  $l^{th}$  existing primary lightpath uses edge  $(i, j)$ , the constant  $a_{ij}^l = 1$ . Equation (5.30) ensures that, if  $a_{ij}^l = 1$ , then either  $X_{ij}^k$  or  $\gamma_{ih}^k$  must be equal to 0.

As we have specified earlier, a translucent lightpath is permitted to have cycles in its physical route, with the restriction that, whenever two segments of a lightpath share a fiber, they must be assigned distinct channel numbers. Equations (5.31) and (5.32) enforce this restriction for the primary and the backup lightpaths respectively.

As we have already mentioned, if a translucent lightpath enters a node with 3R regenerator capability, it does not necessarily mean that we have to use 3R regeneration at that node. To achieve this, each node with 3R regeneration capability should be viewed as a virtual pair of nodes - one with 3R regeneration facility and one without. Both nodes in this virtual pair shares

the same input and output fibers. It may be verified that if a lightpath enters a node with 3R regenerator capability and does not need 3R regeneration, in the solution computed by our formulation, the lightpath will enter the node in the virtual pair with no facility for 3R regeneration.

### 5.3.2 DSLA-2: A Fast ILP Formulation

In formulation DSLA-2, the idea is to limit the search for the routes for any segment of the primary (backup) lightpath to pre-determined edge-disjoint paths. For every pair  $(x, y)$  of nodes, such that it is possible to go from  $x$  to  $y$  without exceeding the optical reach  $r$ , using at least 2 edge-disjoint paths, we pre-compute, if possible,  $\bar{m}$  edge-disjoint paths from  $x$  to  $y$ . Here  $\bar{m}$  is a small number, fixed in advance. If  $\bar{m}$  edge-disjoint paths from  $x$  to  $y$  do not exist, we pre-compute as many edge-disjoint paths from  $x$  to  $y$  as possible. We will use  $\mathbb{R}$  to denote the set of all such pre-computed paths. For the rest of this section, we will use the abbreviation PSTP (for Pre-computed Set of Transparent Paths) to refer  $\bar{m}$  pre-computed edge-disjoint transparent paths between a node-pair.

In response to a new request, to establish a new pair of lightpaths (one primary and one backup) from the source node  $S$  to the destination node  $D$  of the request, one of the following two situations may arise.

1. We may find a PSTP from  $S$  to  $D$  in the set of all PSTP's,  $\mathbb{R}$ .
2. We may not find a PSTP from  $S$  to  $D$  in the set  $\mathbb{R}$ .

For the first situation, formulation DSLA-2 has just to select two routes from the PSTP from  $S$  to  $D$ , one for the primary lightpath and another for

the backup lightpath. Both these lightpaths would be transparent lightpaths that do not need any regeneration.

The second situation is a little complicated. In this case, since we do not have a PSTP from  $S$  to  $D$  in  $\mathbb{R}$ , the pair of lightpaths (primary and backup) to be established from  $S$  to  $D$  must be translucent lightpaths, each involving one or more regenerator node(s). Following the observations in Section ??, DSLA-2 first selects a subset  $\mathbb{P}$  of PSTP's from the set  $\mathbb{R}$ , which we may call the set of PSTP's of interest. PSTP's in  $\mathbb{P}$  will be those PSTP's which are a) from  $S$  to a regenerator, b) from one regenerator to another regenerator, and c) from a regenerator to  $D$ . For each PSTP  $x \Rightarrow y$  in  $\mathbb{P}$ ,  $x$  will be either the source node  $S$  or any regenerator node, and  $y$  will be either the destination node  $D$  or any other regenerator node.

Each PSTP  $x \Rightarrow y$  in  $\mathbb{P}$  will serve as a variable in our formulation DSLA-2. Both the new primary lightpath and the new backup lightpath from  $S$  to  $D$  uses the same PSTP's  $S \Rightarrow R_k \Rightarrow R_p \Rightarrow \dots \Rightarrow R_q \Rightarrow D$ , involving the same regenerators  $R_k, R_p, \dots, R_q$ . Once a route from  $S$  to  $D$  is determined, each PSTP  $x \Rightarrow y$  in the route will correspond to a segment in the primary as well as a segment in the backup lightpath. DSLA-2 will select two routes from PSTP  $x \Rightarrow y$  – one for a segment of the new primary lightpath and another for a segment for the new backup lightpath.

In DSLA-2, in addition to the symbols used in DSLA-1, we will use  $n_p^g$  to denote the number of fibers in the  $g^{th}$  route of PSTP  $p$ . The starting (terminating) node of PSTP  $p$  will be specified by  $s(p)(t(p))$ . Here  $A_p$ ,  $\mathbb{W}_p^{gq}$ ,  $\mathbb{Z}_p^{gq}$ ,  $\mathbb{X}_p^g$ ,  $\mathbb{Y}_p^g$  denote binary variables. If the PSTP  $p$  is selected to handle the new request for communication,  $A_p = 1$ ; otherwise  $A_p = 0$ . If the  $g^{th}$  pre-computed route is selected to realize the segment of the new primary (backup)

lightpath corresponding to PSTP  $p \in \mathbb{P}$ ,  $\mathbb{X}_p^g(\mathbb{Y}_p^g) = 1$ ; otherwise  $\mathbb{X}_p^g(\mathbb{Y}_p^g) = 0$ . If the new primary(backup) lightpath uses channel  $q$  in the  $g^{th}$  route of PSTP  $p$ ,  $\mathbb{W}_p^{gq}(\mathbb{Z}_p^{gq}) = 1$ ; otherwise  $\mathbb{W}_p^{gq}(\mathbb{Z}_p^{gq}) = 0$ .

In formulation DSLA-2,  $\alpha_{ij}^{pg}$ ,  $\kappa_{qg}^{ph}$ ,  $\eta_p^g$ ,  $\theta_p^g$ ,  $b_{ij}^{lh}$  are constants. If the  $g^{th}$  route of PSTP  $p$  includes physical edge  $(i, j)$ ,  $\alpha_{ij}^{pg} = 1$ ; otherwise  $\alpha_{ij}^{pg} = 0$ . If the  $h^{th}$  route of PSTP  $p$  and the  $g^{th}$  route of PSTP  $q$  share some edge(s),  $\kappa_{qg}^{ph} = 1$ ; otherwise  $\kappa_{qg}^{ph} = 0$ . The set of available channel numbers that can be used to set up a new primary lightpath using the route  $g$  of PSTP  $p$  is  $\eta_p^g$ . The set of channel numbers used by the existing primary lightpaths using one or more edges in route  $g$  of PSTP  $p$  will be denoted by  $\theta_p^g$ . The channel number used by the  $l^{th}$  existing backup lightpath in PSTP  $p$  will be denoted by  $\omega_p^l$ . Here  $\lambda_{ph}^{gl}$  denotes a non-negative continuous variable for the new backup path, whose values are restricted by the constraints, such that, if the  $g^{th}$  route for PSTP  $p$  shares an edge with the segment  $h$  of the  $l^{th}$  backup lightpath, and the channel number  $\mathbb{Z}_p^{gq}$  matches with  $\omega_p^l$  then  $\lambda_{ph}^{gl} = 1$ ; otherwise  $\lambda_{ph}^{gl} = 0$ . If the segment  $h$  of the  $l^{th}$  existing backup lightpath uses edge  $(i, j)$ ,  $b_{ij}^{lh} = 1$ ; otherwise  $b_{ij}^{lh} = 0$ .

The formulation is given as:

$$\text{minimize } M \cdot \sum_{p \in \mathbb{P}} A_p + \sum_{p \in \mathbb{P}} \left( \sum_{g^1=1}^{\bar{m}} n_p^{g^1} \cdot \mathbb{X}_p^{g^1} + \sum_{g^2=1}^{\bar{m}} n_p^{g^2} \cdot \mathbb{Y}_p^{g^2} \right) \quad (5.33)$$

subject to:

1. Each PSTP, selected to handle the new request for communication, must



satisfy the flow balance equations.

$$\sum_{p:s(p)=i} A_p - \sum_{p:t(p)=i} A_p = \begin{cases} 1 & \text{if } i = S, \\ -1 & \text{if } i = D, \\ 0 & \text{otherwise.} \end{cases} \quad (5.34)$$

2. For each PSTP, selected to handle the new request for communication, there must exist, through the physical topology, one route for the corresponding segment of the primary lightpath, and one route for the corresponding segment of the backup lightpath

$$\sum_{g=1}^{\bar{m}} \mathbb{X}_p^g = A_p : \quad \forall p \in \mathbb{P}. \quad (5.35)$$

$$\sum_{g=1}^{\bar{m}} \mathbb{Y}_p^g = A_p : \quad \forall p \in \mathbb{P}. \quad (5.36)$$

3. The primary lightpath segment, corresponding to each selected PSTP, must be assigned exactly one channel number, not used by any existing primary or backup lightpath that shares any fiber in the path used by the new segment.

$$\sum_{q:q \in \eta_p^g} \mathbb{W}_p^{gq} = \mathbb{X}_p^g : \quad \forall p \in \mathbb{P}, \quad (5.37)$$

$$\forall g, 1 \leq g \leq \bar{m}.$$

4. The backup lightpath segment, corresponding to each selected PSTP, must not use a channel number assigned to an existing primary lightpath

that shares any fiber in the path used by the new segment.

$$\sum_{q:q \notin \theta_p^g} \mathbb{Z}_p^{gq} = \mathbb{Y}_p^g : \quad \forall p \in \mathbb{P}, \quad (5.38)$$

$$\forall g, 1 \leq g \leq \bar{m}.$$

5. The route used by each segment of the new primary lightpath must be edge-disjoint with respect to the route used by each segment of the new backup lightpath.

$$\mathbb{X}_{p^1}^{g^1} + \mathbb{Y}_{p^2}^{g^2} \leq 1 : \quad \kappa_{p^2, g^2}^{p^1, g^1} = 1, \quad (5.39)$$

$$\forall p^1, p^2 \in \mathbb{P},$$

$$\forall g^1, g^2, 1 \leq g^1, g^2 \leq \bar{m}.$$

6. A segment of the new backup lightpath may share a channel as well as a fiber with a segment of an existing backup lightpath, only if the new primary lightpath is edge-disjoint with respect to the primary lightpath corresponding to that existing backup lightpath. In a way very similar to equations (5.27) - (5.29) of formulation DSLA-1, equations (5.40) - (5.42) ensure that the continuous variable  $\lambda_{ph}^{gl}$  has a value 1, if and only if the  $g^{th}$  pre-computed route for segment  $p$  of the new backup lightpath shares some edge  $(i, j)$  as well as the channel number that are also used by segment  $h$  of the  $l^{th}$  existing backup lightpath; otherwise,  $\lambda_{ph}^{gl}$  has a value 0. Whenever  $\lambda_{ph}^{gl} = 1$ , equation (5.43) ensures that the new primary lightpath does not share any edge in the physical topology with the  $l^{th}$  existing primary lightpath.

$$\begin{aligned} \mathbb{Z}_p^{g\omega^l} + b_{ij}^{lh} \cdot \alpha_{ij}^{pg} \cdot \mathbb{Y}_p^g - \lambda_{ph}^{gl} \leq 1 : \quad & \forall g, 1 \leq g \leq \bar{m}, \\ & \forall p, h \in \mathbb{P}, \\ & \forall l, 1 \leq l \leq \mathcal{L}. \end{aligned} \quad (5.40)$$

$$\begin{aligned} \lambda_{ph}^{gl} - \mathbb{Z}_p^{g\omega^l} \leq 0 : \quad & \forall g, 1 \leq g \leq \bar{m}, \\ & \forall p, h \in \mathbb{P}, \\ & \forall l, 1 \leq l \leq \mathcal{L}. \end{aligned} \quad (5.41)$$

$$\begin{aligned} \lambda_{ph}^{gl} - \sum_{ij: b_{ij}^{lh}=1} \alpha_{ij}^{pg} \cdot \mathbb{Y}_p^g \leq 0 : \quad & \forall g, 1 \leq g \leq \bar{m}, \\ & \forall p, h \in \mathbb{P}, \\ & \forall l, 1 \leq l \leq \mathcal{L}. \end{aligned} \quad (5.42)$$

$$\begin{aligned} \alpha_{ij}^l + \alpha_{ij}^{pg} \cdot \mathbb{X}_p^g + \lambda_{th}^{kl} \leq 2 : \quad & \forall p, t, h \in \mathbb{P}, \\ & \forall g, k, 1 \leq g, k \leq \bar{m}, \\ & \forall l, 1 \leq l \leq \mathcal{L}. \end{aligned} \quad (5.43)$$

7. If the routes used by any pair of segments of the new primary (backup) share a fiber, the pair of primary (backup) segments must be assigned distinct channel numbers.

$$\begin{aligned} \mathbb{W}_{p^1}^{g^1q} + \mathbb{W}_{p^2}^{g^2q} \leq 1 : \quad & \forall q \in Q, \\ & \kappa_{p^2g^2}^{p^1g^1} = 1, \\ & \forall p^1, p^2 \in \mathbb{P}, \\ & \forall g^1, g^2, 1 \leq g^1, g^2 \leq \bar{m}. \end{aligned} \quad (5.44)$$

$$\begin{aligned}
\mathbb{Z}_{p^1}^{g^1q} + \mathbb{Z}_{p^2}^{g^2q} &\leq 1 : \quad \forall q \in Q, \\
\kappa_{p^2}^{p^1g^1} &= 1, \\
\forall p^1, p^2 &\in \mathbb{P}, \\
\forall g^1, g^2, 1 &\leq g^1, g^2 \leq \overline{m}.
\end{aligned} \tag{5.45}$$

### 5.3.2.1 Justification for the Formulation DSLA-2

The objective function of DSLA-2 denotes the same function used in DSLA-1 but uses a slightly different formula due to the way we have formulated DSLA-2. The objective function is a composite function, involving the total number of regenerators in the primary and the backup lightpaths, and the number of channels in the primary and backup lightpaths. By making  $M$  sufficiently large, we ensure in equation (5.33) that the number of regenerators are minimized first and then the number of physical links are minimized.

Most of the equations in formulation DSLA-2 are self descriptive and the explanations are very similar to the corresponding equations in formulation DSLA-1. Equations (5.40) through (5.43) needs some clarifications. These equations correctly compute the value of  $\lambda_{p,h}^{g,l}$ . If  $\lambda_{p,h}^{g,l} = 1$ , then the new primary lightpath must be edge disjoint with respect to the  $l^{th}$  existing lightpath. If segment  $p$  of the new backup lightpath uses route  $g$  and a channel number that has been already used by segment  $h$  of the  $l^{th}$  existing backup lightpath, then  $\mathbb{Z}_p^{g,\omega_h^l} = 1$ . In that case, equation (5.41) indicates that  $\lambda_{p,h}^{g,l} \leq 1$ . If edge  $(i, j)$  appears in segment  $h$  of existing  $l^{th}$  backup lightpath,  $b_{ij}^{l,h} = 1$ . If edge  $(i, j)$  also appears in the  $g^{th}$  route of the node pair  $p$ ,  $\alpha_{ij}^{pg} = 1$ . If segment  $p$  of the new backup lightpath uses route  $g$ ,  $\mathbb{Y}_p^g = 1$ . Equation (5.40) states that,

in this situation,  $\lambda_{p,h}^{g,l} \geq 1$ . The only value of  $\lambda_{p,h}^{g,l}$  that satisfies both (5.40) and (5.41), in this case, is  $\lambda_{p,h}^{g,l} = 1$ .

If, on the other hand, the new backup path does not share any edge as used by the  $l^{th}$  backup path in any segment, equation (5.42) states that  $\lambda_{p,h}^{g,l} \leq 0$ . Since all variables must be greater than or equal to 0,  $\lambda_{p,h}^{g,l}$  must be 0. If segment  $p$  of the new backup path does not share the same channel number used by segment  $h$  of the  $l^{th}$  backup path,  $\mathbb{Z}_p^{g,\omega_h} = 0$ . In this situation, equation (5.40) states that  $\lambda_{p,h}^{g,l} \geq 0$ . Since (5.42) states that  $\lambda_{p,h}^{g,l} \leq 0$ , the only solution that satisfies both (5.40) and (5.42), is  $\lambda_{p,h}^{g,l} = 0$ .

In summary, the equations (5.40), (5.41) and (5.42) ensure that, for all  $g, 1 \leq g \leq \bar{m}$  and for all  $p, h \in \mathcal{P}$  and for all  $l, 1 \leq l \leq \mathcal{L}$ , the continuous variable  $\lambda_{p,h}^{g,l}$  has a value 1 if and only if

- the new backup lightpath in its  $g^{th}$  route of the segment  $p$ , shares some edge  $(i, j)$  in the physical topology that is also used by the  $l^{th}$  existing backup lightpath in its segment  $h$  and
- the new backup lightpath in the same  $g^{th}$  route of the segment  $p$ , shares a channel number that is also used by the  $l^{th}$  existing backup lightpath in its segment  $h$ .

Otherwise  $\lambda_{p,h}^{g,l}$  has a value 0.

If the  $l^{th}$  existing primary lightpath uses the edge  $(i, j)$ , the constant  $a_{ij}^l = 1$ . Equation (5.43) ensures that, if  $a_{ij}^l = 1$ , then either  $\alpha_{ij}^{pg} \cdot \mathbb{X}_p^g$  or  $\lambda_{p,h}^{g,l}$  must be equal to 0.

As stated earlier, a translucent lightpath is permitted to have cycles in its physical routing with the restriction that whenever two segments of a lightpath share a fiber, they must be assigned distinct channel numbers. Equations

(5.44) and (5.45) enforce this restriction for the primary lightpath and the backup lightpath respectively.

### 5.3.3 Experimental Results

For experiments, we have considered five randomly generated different sizes of networks, with 8, 16, 24, 32 and 40 nodes respectively. We have randomly assigned the number of incoming and outgoing fibers at each node of a network from 3 to 5. We have randomly selected the length of the fiber between any two directly connected nodes between 100 to 1000 km. We have selected the optical reach  $r$  to be 1000 km. We have chosen the number of wavelength channels on the fibers to be 8. For each size of a network that we have considered, we randomly generated 5 sets of topologies. The results shown here are the average values of data from those 5 sets.

To compare the performances of DSLA-1 and DSLA-2, we have carried out the experiments for various network densities, such as, 10, 20, 30, 40 and 50 erlangs. We have conducted our simulations for 500 iterations for each value of the size the network and for each network density, where the requests for data communication arrived at a random rate and had a random holding time.

For the simulations with DSLA-2, we pre-computed three edge-disjoint shortest paths, if possible, through the physical topology between each pair of nodes that does not require a regenerator node. For any given pair of such nodes, if three edge-disjoint paths could not be found, we selected two edge-disjoint shortest paths between the pair, i.e.,  $2 \leq \bar{m} \leq 3$ . The experiments were carried out on a Sun Fire X2200 M2 Server [76], using ILOG CPLEX version 11.1 [36].

Figure 5.8 compares the percentage of blocked requests using DSLA-1 and

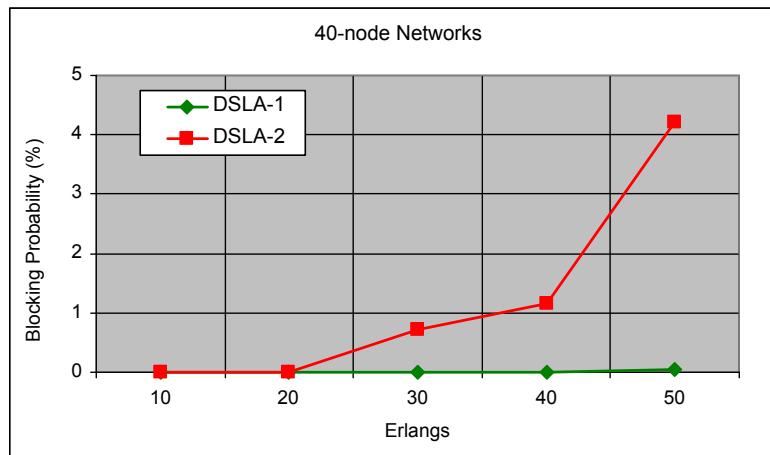


Figure 5.8: Blocking probabilities using DSLA-1 and DSLA-2 for 40-node networks.

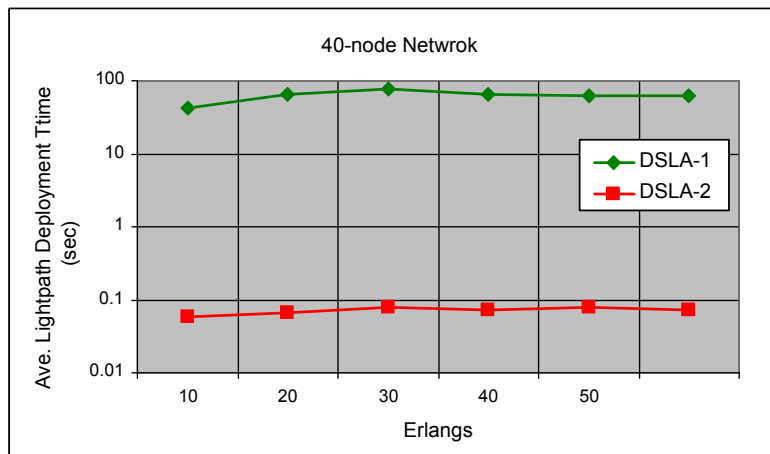


Figure 5.9: Individual lightpath deployment time using DSLA-1 and DSLA-2 for 40-node networks.

DSLA-2 for a 40-node network at different erlang values. As expected, the percentage of blocked requests is higher with DSLA-2. DSLA-1 searches more routes and is often able to find a solution, where DSLA-2 fails. The figure also shows that, although the blocking probability with DSLA-2 is higher, it is well within the acceptable limits (less than 5% for 50 erlangs). However, DSLA-2 can produce results much faster than the DSLA-1 (on an average

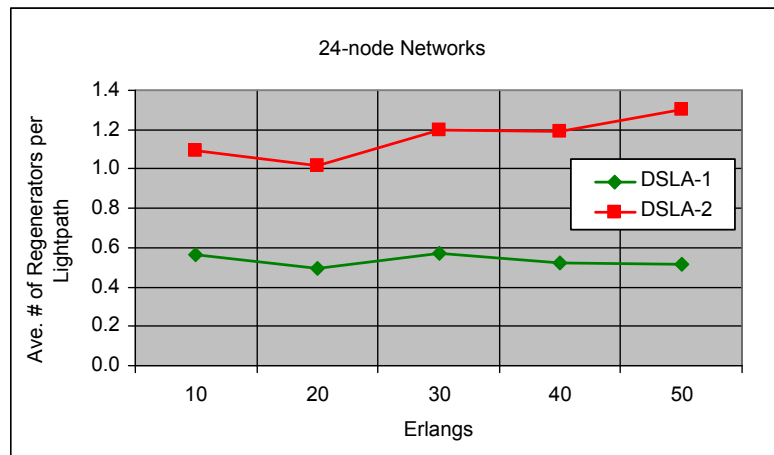


Figure 5.10: Number of regenerators required by DSLA-1 and DSLA-2 for 24-node networks.

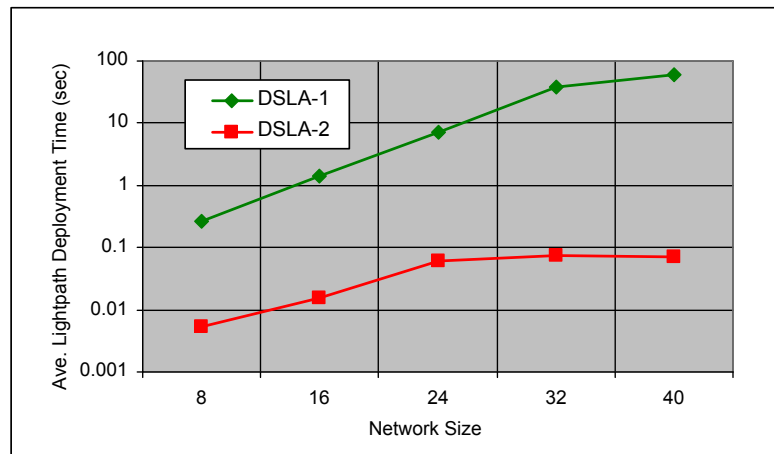


Figure 5.11: Average lightpath deployment time using DSLA-1 and DSLA-2 for different networks.

100 times faster). The comparison of the execution time, using a logarithmic scale, is shown in Figure 5.9 for 40-node networks. The figure shows that, given a network topology, the time needed to search for a path and deploying a lightpath does not vary significantly with the congestion of the network, which is what we have expected. We have observed almost identical results for other networks as well.



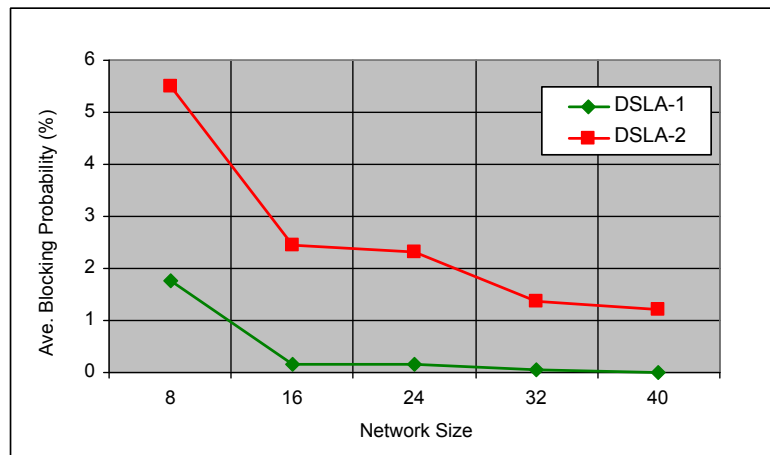


Figure 5.12: Average blocking probability using DSLA-1 and DSLA-2 for different networks.

In Figure 5.10, we have compared the number of regenerators used by the two formulations for 24-node networks, for the same set of successful requests. As expected, the number of regenerators required to establish a lightpath (both primary and backup) does not vary significantly with traffic congestion in the network. However, as shown in Figure 5.10, DSLA-1 requires less regenerators than DSLA-2 in all cases.

Figure 5.11 compares the average time required to establish lightpaths in different size networks using DSLA-1 and DSLA-2. The figure clearly shows that DSLA-2 can deploy a lightpath much faster than DSLA-1. For all the cases DSLA-2 needs, in an average, less than 100 ms, whereas DSLA-1 needs around 10,000 ms. In the case of dynamic lightpath allocation, it is very important to obtain a solution very quickly. Therefore, if the performance is acceptable, DSLA-2 can be effectively used to meet the requirements of the dynamic environment.

Figure 5.12 shows the average blocking probability using DSLA-1 and

DSLAs-2 for networks of different size. As expected, the blocking probability using DSLAs-2 is always higher than with DSLAs-1. However, the figure also shows that the blocking probability of DSLAs-2 is well within the acceptable limits (less than 5%). With such efficient performance in terms of time required, we expect that this small increase in the blocking probability will be acceptable to the industry.

## 5.4 Static Lightpath Allocation in Translucent Networks

The RPP and RRP problems studied by the research community primarily deal with dynamic lightpath allocation [3] and typically do not take into account the fact that the number of channels on a fiber is limited. We have studied static lightpath allocation [3] in translucent networks and have pointed out that this problem is significantly different from the RRP and RPP problems that have been studied.

An example of a long haul network with distances between the nodes in km is shown in figure 5.13. If the optical reach is 2000 km, it is clear that an optical signal from node *A* cannot reach node *D* without 3R-regeneration. However, communication between *A* and *D* can be established by placing a regenerator either at *B* or *C*. It may be verified that a minimum of 2 regenerators are needed for this network so that every node can communicate with every other node in the network. One solution is to place these regenerators at nodes *B* and *D*. When designing a translucent network, the RPP problem must be solved before tackling the problem of lightpath allocation.

In order to regenerate an optical signal we need, at a minimum, a receiver and a transmitter [86]. The simplified model for determining the “cost” of

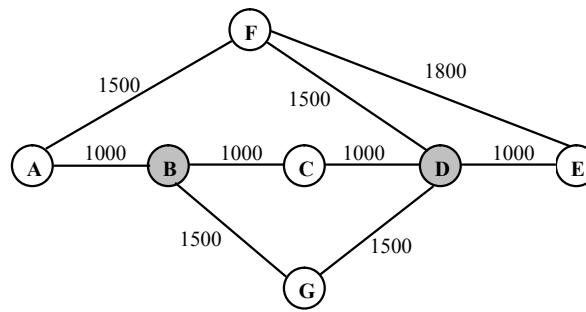


Figure 5.13: A wide area translucent network with distances between the nodes in km.

regeneration at any given node having 3R regeneration facility is the cost of a receiver and the cost of a transmitter, for each lightpath undergoing regeneration at that node<sup>6</sup>. Let the network shown in Figure 5.13 have an optical reach of 2000 km. The regenerators are at nodes  $B$  and  $D$ , as prescribed by any standard RPP algorithm. We consider the problem of establishing three lightpaths i) from  $A$  to  $E$ , ii) from  $B$  to  $E$ , and iii) from  $G$  to  $E$ . The following situations illustrate the limitations of standard RRP and RPP when carrying out static lightpath allocation in this network:

**Situation 1:** Let each fiber in the network accommodate two channels,  $c_1$  and  $c_2$ . If we use the RRP approach [4, 67, 85, 87], the lightpath  $A$  to  $E$  must undergo regeneration at nodes  $B$  and  $D$ , and use the route  $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$ . Similarly, the lightpath from  $B$  to  $E$  ( $G$  to  $E$ ) must undergo regeneration at node  $D$  and use the route  $B \rightarrow C \rightarrow D \rightarrow E$  ( $G \rightarrow D \rightarrow E$ ). As all three routes share the edge  $D \rightarrow E$ , this is an *invalid* solution, since the edge  $D \rightarrow E$  allows only two channels. However the network is capable of supporting these three lightpaths, if we place an *additional* regenerator at node  $F$  and, for the lightpath from  $A$  to  $E$ , use the route  $A \rightarrow F \rightarrow E$ , instead

<sup>6</sup>For static RWA, fixed frequency transmitters and receivers offer the most economic solution.

of the route  $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$ .

**Situation 2:** Let each fiber in the network support three channels,  $c_1$ ,  $c_2$  and  $c_3$ . It would now be possible to use route  $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$  ( $B \rightarrow C \rightarrow D \rightarrow E$ , and  $G \rightarrow D \rightarrow E$ ) for the lightpath from  $A$  to  $E$  (respectively  $C$  to  $E$ , and  $G$  to  $E$ ) and use the scheme of placing regenerators shown in Figure 5.13. The total number of lightpaths undergoing regeneration at node  $B$  ( $D$ ) is 1 (3), so that the cost for regeneration is  $1 + 3 = 4$ . However, this cost for regeneration may be reduced to 3 if the lightpath from  $A$  to  $E$  undergoes regeneration at node  $C$  instead of nodes  $B$  and  $D$ .

In summary the design objective for static RWA should be the minimization of the total number of regenerations that the lightpaths need to undergo, rather than the minimization of the number of nodes having regeneration facility.

In order to determine, for each  $k \in K$ , a feasible RWA for a translucent lightpath from  $S^k$  to  $D^k$ , our algorithm must find i) the actual route through the physical topology, to be used by the translucent lightpath from  $S^k$  to  $D^k$ , ii) the node(s) on the route from  $S^k$  to  $D^k$  where the lightpath has to undergo regeneration, and iii) the channel to be used for each segment of the lightpath.

In our first approach we have used an Integer Linear Program (ILP) formulation (we called it SLA for Static Lightpath Allocation), which finds an optimal solution to the problem. For larger networks we have proposed a two-step heuristic. The first step of the heuristic uses another Integer Linear Program formulation (we called it ILPH for ILP for Heuristic), to determine a *preliminary* route for each lightpath to be set up. The second step uses a search algorithm, that takes as its input, the preliminary route obtained in Step 1, and solves the RWA problem where the actual route used by a lightpath will differ from the preliminary route determined in Step 1, if channel

assignment on any segment on the preliminary route is not possible.

For SLA (ILPH), the process of finding a route (preliminary route) for a lightpath  $k \in K$ , may be viewed as equivalent to a multi-commodity network flow problem over the physical topology, where a single unit of commodity  $k \in K$ , distinct from all other commodities, is being shipped from  $S^k$  to  $D^k$  using a non-bifurcated route. As such, for the rest of this chapter, we will use the words lightpath and commodity interchangeably.

#### 5.4.1 SLA: An ILP Formulation for Optimal Solution

In the ILP formulation for the optimal static lightpath allocation (SLA) in translucent networks, we have used the following symbols. Here  $N$  denotes the set of all end-nodes in the physical network.  $E$  denotes the set of all pairs  $(i, j)$  such that there is a directed edge  $i \rightarrow j$  from node  $i$  to node  $j$  in the physical topology, representing a fiber from node  $i$  to node  $j$ . The physical topology is represented by a graph  $G$ , such that  $G = \{N, E\}$ .  $K$  is the set of source-destination pairs  $(S^k, D^k)$  of all lightpaths for which RWA is to be carried out.  $Q$  represents the set of channels on each fiber. The optical reach of the network is denoted by  $r$ .  $S^k$  ( $D^k$ ) denotes the source (destination) node for commodity  $k \in K$ . The length of the physical edge  $(i, j) \in E$  is  $d_{ij}$ .  $\varphi_{ij}^k$  is a binary flow variable for each commodity  $k \in K$  over an edge  $(i, j) \in E$  in the physical topology. Commodity  $k$  uses edge  $i \rightarrow j$ ,  $\varphi_{ij}^k = 1$ , otherwise  $\varphi_{ij}^k = 0$ .  $\beta_i^k$  is a binary variable for each node  $i \in N$ . If commodity (lightpath)  $k$  is regenerated at node  $i$ ,  $\beta_i^k = 1$ , otherwise  $\beta_i^k = 0$ .  $w_{ij}^{kq}$  denotes a binary variable for each channel  $q \in Q$ , each commodity  $k \in K$  and each edge  $(i, j) \in E$ , such that, if commodity  $k$  uses channel  $q$  on edge  $(i, j)$ ,  $w_{ij}^{kq} = 1$ , otherwise  $w_{ij}^{kq} = 0$ .  $M$  is a large integer number.

Here  $v_i^k$  is a continuous variable for each node  $i \in N$  in the physical topology and each commodity  $k \in K$ . The value of  $v_i^k$  is restricted by the following constraint. If there is a physical edge between node  $i$  and node  $j$ , such that  $i \rightarrow j \in E$  and the nodes  $i$  and  $j$  are on the route used by the commodity  $k$ , the value of  $v_j^k$  depends upon the value of  $v_i^k$  and the distance  $d_{ij}$  between node  $i$  and node  $j$ , and is given by  $v_j^k = v_i^k + d_{ij}$ , provided that  $j \neq S^k$  and  $\beta_j^k = 0$ . If  $j = S^k$  or  $\beta_j^k = 1$ , then  $v_j^k = 0$ .

The formulation is given as:

$$\text{minimize } \sum_{k \in K} \sum_{i \in N} M \cdot \beta_i^k + \sum_{k \in K} \sum_{(i,j) \in E} \varphi_{ij}^k \quad (5.46)$$

subject to:

1. Each edge selected for a commodity must satisfy the flow balance equations.

$$\sum_{j:(i,j) \in E} \varphi_{ij}^k - \sum_{j:(j,i) \in E} \varphi_{ji}^k = \begin{cases} 1 & \text{if } i = S^k, \\ -1 & \text{if } i = D^k, \\ 0 & \text{otherwise.} \end{cases} \quad (5.47)$$

Constraint (5.47) must be satisfied  $\forall i \in N$  and  $\forall k \in K$ .

2. The route corresponding to a commodity cannot contain a cycle.

$$\sum_{i:(i,j) \in E} \varphi_{ij}^k \leq 1; \quad \forall j \in N, \forall k \in K \quad (5.48)$$

3. The length of the fibers along the route of a transparent segment cannot exceed the optical reach  $r$  of the network.

$$v_i^k + d_{ij}\varphi_{ij}^k - r(1 - \varphi_{ij}^k + \beta_i^k) \leq v_j^k; \quad \forall (i, j) \in E, \forall k \in K \quad (5.49)$$

$$v_i^k + d_{ij}\varphi_{ij}^k \leq r; \quad \forall (i, j) \in E, \forall k \in K \quad (5.50)$$

$$0 \leq v_j^k \leq r(1 - \beta_i^k); \quad \forall j \in N, \forall k \in K \quad (5.51)$$

$$v_{S^k}^k = 0; \quad \forall k \in K \quad (5.52)$$

4. Each commodity must use only one distinct channel on each fiber in its path and a given channel can be allocated to at most one commodity.

$$\sum_{q \in Q} w_{ij}^{kq} = \varphi_{ij}^k; \quad \forall (i, j) \in E, \forall k \in K \quad (5.53)$$

$$\sum_{k \in K} w_{ij}^{kq} \leq 1; \quad \forall (i, j) \in E, \forall q \in Q \quad (5.54)$$

5. The number of lightpaths using an edge cannot exceed the number of channels on a fiber.

$$\sum_{k \in K} \varphi_{ij}^k \leq |Q|; \quad \forall (i, j) \in E \quad (5.55)$$

6. Each transparent segment must satisfy the wavelength continuity constraint.

$$\sum_{i:(i,j) \in E} w_{ij}^{kq} + \beta_i^k \geq \sum_{i:(j,i) \in E} w_{ji}^{kq}; \quad (5.56)$$

$$\sum_{i:(i,j) \in E} w_{ij}^{kq} - \beta_i^k \leq \sum_{i:(j,i) \in E} w_{ji}^{kq}; \quad (5.57)$$

Both constraints (5.56) and (5.57) must be satisfied  $\forall j \in N$  such that,  $j \neq S^k, j \neq D^k, \forall k \in K$  and  $\forall q \in Q$ .

#### 5.4.1.1 Justification for the Formulation SLA

Our objective for the SLA formulation is to minimize a composite function that has two components – i) the total number of regenerations, and ii) the total number of edges, used by all the lightpaths. Here  $M$  is a suitably selected large integer, so that the first component always dominates the second. This means that the primary objective is to minimize the total number of regenerations needed, and the secondary objective is to reduce the total number of edges used, by the lightpaths. Equation (5.47) is the standard flow conservation requirements used in network flow programming [1]. Equation (5.48) ensures that a lightpath does not pass through a node more than once, preventing any possible cycle in the route.

Equations (5.49), (5.50), (5.51) and (5.52) ensure that, if node  $j$  lies on the path of the lightpath corresponding to  $k$ , then,

1. if  $j$  is not a regenerator,
  - $v_j^k$  gives the length of the route from the last regenerator before node  $j$  (or from the source node  $S^k$ , if there is no such regenerator before  $j$ ), to  $j$ .



- the value of  $v_j^k$  never exceeds the optical reach  $r$ .
2. if  $j$  is a regenerator node (i.e  $\beta_i^k = 1$ ), or  $j$  is the source node  $S^k$ , then  $v_j^k = 0$ .

To achieve these requirements, these equations, if needed, force the value of  $\beta_i^k$  to be 1, so that the lightpath undergoes regeneration at node  $j$ . If, in equation (5.49),  $\varphi_{ij}^k = 0$ , meaning that the lightpath  $k$  has not used the edge  $i \rightarrow j$ , equation (5.49) becomes  $v_i^k - r(1 + \beta_i^k) \leq v_j^k$  - a redundant constraint. If, the lightpath  $k$  has used the edge  $i \rightarrow j$ , then  $\varphi_{ij}^k = 1$  and the equation (5.49) becomes  $v_i^k + d_{ij} - r \cdot \beta_i^k \leq v_j^k$ . Since we are minimizing the sum of the values of  $\beta_i^k$ , the solver forces  $\beta_i^k = 0$ , if possible. If the solver can make  $\beta_i^k = 0$ , then  $v_i^k + d_{ij} \leq v_j^k$ . In view of equation (5.50), the solver uses the minimum value of  $v_j^k$ . In other words, if the solver can make  $\beta_i^k = 0$ ,  $v_j^k$  is forced to be  $v_i^k + d_{ij}$ . When the solver is forced to make  $\beta_i^k = 1$ , then equation (5.51) ensures that  $v_j^k = 0$ , and equation (5.49) becomes  $v_i^k + d_{ij} - r \leq 0$  - a redundant constraint.

The LHS of equation (5.53) gives the total number of channels allotted to commodity  $k$  on edge  $i \rightarrow j$ . This total is 1 (0) if the lightpath corresponding to  $k$  uses (does not use) edge  $i \rightarrow j$ . In other words, equation (5.53) ensures that each lightpath is assigned exactly only one channel on each edge in its path and, if a lightpath does not use an edge, no channel is allotted to it on that edge. Equation (5.54) ensures that a channel  $q \in Q$  on any edge  $(i, j) \in E$  is assigned to at most one lightpath. Equation (5.55) ensures that the total number of lightpaths through any edge never exceeds the total number of channels on an edge.

If the lightpath corresponding to  $k$  has not undergone regeneration at node

$j$ ,  $\beta_i^k = 0$ , so that equations (5.56) and (5.57) boil down to  $\sum_{i:(i,j) \in E} w_{ij}^{kq} = \sum_{i:(j,i) \in E} w_{ji}^{kq}$ . The value of  $\sum_{i:(i,j) \in E} w_{ij}^{kq} \left( \sum_{i:(j,i) \in E} w_{ji}^{kq} \right)$  is 1 only if the lightpath entering (leaving) node  $j$  uses channel  $q$ . In this case, equations (5.56) and (5.57) ensure that the channel assigned to lightpath  $k$ , when it enters  $j$ , is the same as the channel assigned to it when it leaves  $j$ . If, on the other hand,  $\beta_i^k = 1$  (in other words, the lightpath corresponding to  $k$  has undergone regeneration at node  $j$ ), then both equations (5.56) and (5.57) are trivially satisfied, so that the value of  $\sum_{i:(j,i) \in E} w_{ji}^{kq}$  is independent of the value of  $\sum_{i:(i,j) \in E} w_{ij}^{kq}$ .

#### 5.4.2 A Heuristic for Static Lightpath Allocation

Algorithm (6) gives an overview for our proposed heuristic to solve the Static Lightpath Allocation problem for large translucent networks. In this algorithm we have used the symbol  $E_s$  to represent the set of physical edges ( $E_s \subseteq E$ ) that constitutes a segment of a translucent lightpath, having a total length  $\leq r$ . The channel used in a segment is denoted by  $q \in Q$ .

In Step 1 of Algorithm 6, we use function *find\_preferred\_routes* to compute *list\_routes* - a list of the preliminary routes from source  $S^k$  to destination  $D^k$ , for all  $k \in K$ . We choose these preliminary routes, so that the maximum number of routes sharing an edge  $(i, j) \in E$  is as small as possible. We have used an ILP formulation (ILPH), described in Section 5.4.3 to solve this problem.

In Step 2, we sort the list of all the routes in some pre-defined order. We have deployed the lightpaths corresponding to the source destination pairs in  $K$  sequentially, by selecting elements in  $K$  in some order. We have studied

---

**Algorithm 6** Heuristic for RWA in translucent networks.

---

**Input:** Physical topology  $G$ , set of lightpaths  $K$

**Output:** RWA for the set of lightpaths  $K$

```

1:  $list\_routes \leftarrow find\_preferred\_routes(G, K)$ 
2: sort  $list\_routes$ 
3: while ( $route \leftarrow get\_next\_route(list\_routes)$ )  $\neq \emptyset$  do
4:   while ( $E_s \leftarrow get\_segment(route)$ )  $\neq \emptyset$  do
5:      $q \leftarrow assign\_channel(E_s)$ 
6:     if  $q \neq 0$  then
7:        $route \leftarrow update\_route(E_s, q)$ 
8:     else
9:        $(new\_E_s, q) \leftarrow search\_A^*(E_s)$ 
10:       $route \leftarrow update\_route(new\_E_s, q)$ 
11:    end if
12:  end while
13: end while

```

---

different orders (e.g., the longest-route-first, the shortest-route-first or random selection of routes) and have observed that, deploying the lightpaths in the order of longest-route-first, in general, gives the best performance. This is due to the fact that a longer lightpath is likely to need a free channel for more edges in its route, which may not be available if the shorter lightpaths are already established. This observation is in line with [43].

Steps 3 - 12 are repeated for all routes in  $list\_routes$ , in order to deploy a lightpath for each  $k \in K$ . In Step 3, function  $get\_next\_route$  returns, if possible, the next route in  $list\_routes$ . If all routes in  $list\_routes$  have been already considered, the function returns  $\emptyset$ . A typical route returned by this function is from source  $S^k$  to destination  $D^k$ , for some  $k \in K$ . Such a route, traversing, say  $e$  edges, has the form  $(S^k = x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_p \rightarrow x_{p+1} \rightarrow \dots \rightarrow x_{e-1} \rightarrow x_e = D^k)$ , where  $x_i \in N, \forall i, 1 \leq i \leq e$ .

Steps 4 - 11 are repeated as long as all the segments have not been identified for the current route, say from  $S^k$  to  $D^k$ . When we execute Step 4,  $route$

denotes the part of the route from  $S^k$  to  $D^k$  for which we have not identified all the viable segments. For instance, let the segments from  $S^k$  to some node  $x_t$  on the route  $S^k = x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_{t-1} \rightarrow x_t \rightarrow x_{t+1} \rightarrow \dots \rightarrow x_e = D^k$  be identified already. The remaining part of the route from  $S^k$  to  $D^k$  is  $x_t \rightarrow x_{t+1} \rightarrow \dots \rightarrow x_p \rightarrow x_{p+1} \rightarrow \dots \rightarrow x_{e-1} \rightarrow x_e = D^k$ . In Step 4, function *get\_segment* tentatively identifies the first segment, by determining the furthest possible location of the first regenerator on *route*. We note that there is no guarantee that this is a viable segment, since a valid channel assignment is not guaranteed for this segment. Function *get\_segment* returns the route starting with the first node of the current route and terminating with the node denoting the location of the first regenerator. The function *get\_segment* identifies node  $x_p$ , such that the length of the route from  $x_t$  to  $x_p$  ( $x_{p+1}$ ) is less than or equal to (greater than) the optical reach  $r$ . Node  $x_p$  is the location of the next regenerator, if the route from  $x_t$  to  $x_p$  can be a viable segment. Function *get\_segment* returns the route  $x_t \rightarrow x_{t+1} \rightarrow \dots \rightarrow x_p$ . There is a special case where the length of the route from  $x_t$  up to  $x_e = D^k$  is less than or equal to  $r$ . In this special case, no regenerator is needed and the function returns the route  $x_t \rightarrow x_{t+1} \rightarrow \dots \rightarrow x_{e-1} \rightarrow x_e = D^k$ . After determining the last segment in the route from  $S^k$  to  $D^k$ , if the function is called again, it returns  $\emptyset$ .

In Step 5, the idea is to attempt a channel assignment to every edge in  $E_s$  to check whether  $E_s$ , determined in Step 4, represents a viable segment. Function *assign\_channel* assigns, if possible, a channel  $q$  for  $E_s$  such that, for each edge on  $E_s$ , the channel  $q$  is not used by any lightpath deployed so far. If *assign\_channel* is successful it returns the selected channel number, otherwise it returns 0. If *assign\_channel* is successful (i.e.,  $q \neq 0$  in Step 6), then we

carry out Step 7; otherwise we carry out Step 9, followed by Step 10.

In Step 7, function *update\_route* updates a database with the information describing the segment, including the channel number  $q$ , assigned to the segment, the route corresponding to  $E_s$  and, if the node is not the destination  $D^k$ , mark the last node of the segment as a regenerator node for this lightpath. In Step 9, we use *search\_A\**, a best first search using the  $A^*$  algorithm [32], to identify *new\_Es* - a viable segment starting from  $x_t$  and terminating at the furthest node  $x_e$  lying on  $E_s$ , for which a channel  $q$  can be assigned. We note that the route from  $x_t$  to  $x_e$  has no relation to the route in  $E_s$  from  $x_t$  to  $x_p$ . In *search\_A\**, we are searching for the shortest path from  $x_t$  to  $x_p$  for which a valid channel assignment can be done. Our heuristic cost from any node, say  $n \in N$  to  $x_p$  is the length of the shortest path from  $n$  to  $x_p$ , which we have computed in a separate step, using the Dijkstra's algorithm [1] before using Algorithm 1. Since this heuristic cost is less than or equal to the actual cost of any path from  $n$  to  $x_p$ , the heuristic is an admissible heuristic and will find the "best" path to  $x_p$ , if such a path exists. When heuristic search terminates, either we have reached  $x_p$  using a path for which a channel assignment can be done, or no such path exists. Step 10 is just like Step 7, using *new\_Es* and the corresponding channel  $q$ .

### 5.4.3 ILPH: An ILP Formulation used in the Heuristic

The ILP formulation for the function *find\_preferred\_routes* can be given as follows:

$$\text{minimize } \Psi_{max} + \sum_{k \in K} y^k \quad (5.58)$$

subject to:

1. Each edge selected for a lightpath must satisfy the flow balance equations.

$$\sum_{j:(i,j) \in E} \varphi_{ij}^k - \sum_{j:(j,i) \in E} \varphi_{ji}^k = \begin{cases} 1 & \text{if } i = S^k, \\ -1 & \text{if } i = D^k, \\ 0 & \text{otherwise.} \end{cases} \quad (5.59)$$

Constraint (5.59) must be satisfied  $\forall i \in N$  and  $\forall k \in K$ .

2. The route corresponding to a lightpath cannot contain a cycle.

$$\sum_{i:(i,j) \in E} \varphi_{ij}^k \leq 1 : \quad \forall j \in N, \forall k \in K \quad (5.60)$$

3.  $\Psi_{max}$  cannot exceed the number of channels per fiber.

$$\Psi_{max} \leq |Q| \quad (5.61)$$

4. The number of lightpaths through an edge must be less than or equal to  $\Psi_{max}$ .

$$\sum_{k \in K} \varphi_{ij}^k \leq \Psi_{max} : \quad \forall (i, j) \in E. \quad (5.62)$$

5. Determine the number of regenerators  $y^k$  needed for the lightpath  $k \in K$ .

$$y^k \geq \lfloor (\sum_{(i,j) \in E} \varphi_{ij}^k \cdot d_{ij}) / r \rfloor : \quad \forall k \in K. \quad (5.63)$$

In the above formulation we have used the following notations in excess to some of the notations explained in Section 5.4.1. Here  $\Psi_{max}$  denotes the *congestion* of the network, defined by the number of lightpaths flowing through the edge carrying the maximum number of lightpaths.  $y^k$  is an integer variable for commodity  $k \in K$ , denoting the minimum number of segments needed for commodity  $k$ .

#### 5.4.3.1 Justification for the Formulation ILPH

Our objective for ILPH formulation is to find initial routes for all the lightpaths to be deployed. To achieve the objective, the formulation tries to minimize a composite function that has two components. The first component minimizes the overall congestion of the network. The second component tries to select a route for the lightpath such that the estimated number of regenerators needed for the lightpath, is minimized.

Equation (5.59) is the standard flow conservation requirements used in network flow programming [1]. Equation (5.60) ensures that a lightpath can not visit a node more than once, preventing any possible cycle in the route. Equation (5.61) ensures that the congestion of the network, which is the total number of lightpaths in an edge or edges, cannot be more than the number of wavelength channels per fiber. Equation (5.62) ensures that the total number of lightpaths in any edge is always less than or equal to the congestion  $\Psi_{max}$ . Equation (5.63) calculates the estimated minimum number of regenerators needed for each lightpath. For a lightpath  $k \in K$ , it sums up the lengths  $d_{ij}$  of all the edges for which  $\varphi_{ij}^k = 1$ , divide the total distance by the optical reach  $r$  and takes the floor value.

#### 5.4.4 Experimental Results

To compare the performance of the heuristic in Section 5.4.2 to the ILP formulation SLA in Section 5.4.1, we have generated a number of networks, with sizes ranging from 8 to 14 and have run both formulations<sup>7</sup>. To study the performance of the heuristic for medium and large sized networks, we have generated networks with the number of nodes ranging from 20 to 100 nodes. We have randomly chosen the degree of each node to lie in the range 2 to 4 (2 to 5) for networks with 14 nodes or less (20 nodes or more). For networks with 14 nodes or less (20 nodes or more), we have selected the number of channels in each fiber to be 12 (16). If a node-pair is connected by a fiber, we have randomly selected the length of the fiber to lie in the range 400 to 1800 km. We used  $r = 2000$  km.

To study the effect of varying the total number of commodities handled by a network, we have grouped the number of commodities handled by a network into 3 categories and have called them *low*, *medium* and *high densities*. For low (medium, and high) density networks, we have chosen the number of commodities to be 2 (3, and 4) times of the number of nodes in the network. For each size of the networks we studied, we have randomly generated 5 sets of physical topologies. For each density (low, medium or high) we have randomly generated 5 sets of commodities, each set consisting of the requisite number of randomly generated source destination pairs. The results reported in this paper represent the average values of those 75 sets data for each size of networks. We have carried out the experiments on a Sun Fire X2200 M2 [76].

Table 5.1 shows a comparison of times needed to solve the SLA formulation

---

<sup>7</sup>Larger networks takes an unacceptable amount of time using SLA.



Table 5.1: Comparison of SLA and Heuristic Performances

Network Size	Commodities	Regenerations		Channels		Solution Time (sec)	
		SLA	Heuristic	SLA	Heuristic	SLA	Heuristic
8	16	7.8	8.5	29.4	31.2	7.153	0.378
	24	12.0	12.8	44.2	46.9	18.799	0.366
	32	15.9	17.5	58.6	63.0	70.504	0.406
10	20	9.9	10.2	39.2	41.0	26.452	0.294
	30	15.9	16.8	59.6	62.9	103.527	0.310
	40	19.2	20.6	78.0	81.9	236.580	0.351
12	24	14.6	14.8	52.5	54.5	594.598	0.331
	36	21.1	22.5	76.9	79.4	1131.423	0.315
	48	30.4	31.1	104.7	106.9	315.331	0.349
14	28	21.0	22.2	64.7	66.7	506.788	0.318
	42	30.3	34.3	100.7	102.2	411.371	0.359
	56	40.8	44.9	131.6	135.2	1856.786	0.388

Table 5.2: Heuristics Results for Large Networks

Network Size	Commodities	Regenerations	Channels	Solution
		Used	Used	Time (sec)
20	40	26.5	97.8	0.462
	60	38.9	144.7	0.540
	80	52.7	195	0.635
40	80	88.6	247.6	1.156
	120	135.2	378.1	1.644
	160	178.8	501.0	2.370
60	120	160.4	427.1	2.723
	180	237.0	631.7	5.325
	240	318.6	841.5	23.360
80	160	245.7	602.6	5.948
	240	373.4	901.1	34.720
	320	496.9	1201.8	158.525
100	200	317.6	799.3	8.507
	300	477.8	1203.6	102.650
	400	638.2	1592.8	235.506

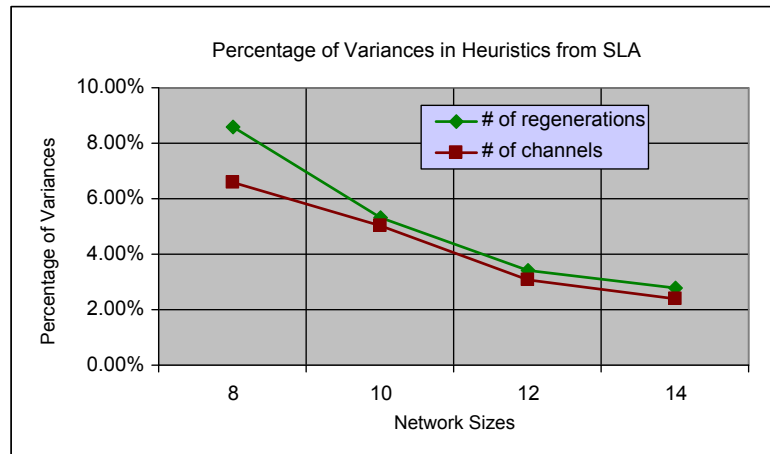


Figure 5.14: Percentage of variances of outputs using Heuristic and SLA

to the time needed for solving using the heuristic, for networks of sizes 8, 10, 12 and 14 nodes. It also shows the average numbers of regenerations and the average number edges per lightpath used by the SLA and the heuristic. Columns 1 (2) in Table 5.1 shows  $|N|$  ( $|K|$ ), the size of the network (the number of commodities).

Table 5.2 shows, for medium and large sized networks, the same parameters as Table 5.1 when the heuristic is used.

Figure 5.14 shows how well our heuristic works by depicting the percentage of variances of the results obtained by the heuristic compared to the optimal results. We find that this variance decreases monotonically, as the size of the network increases. We conclude that our heuristic can produce “near optimal” solutions using only a fraction of the time needed to obtain an optimal solution using SLA.

## Chapter 6

# Conclusions and Future Works

### 6.1 Conclusions

In this Ph.D. dissertation, we have presented novel algorithms to find optimal solutions for different design problems in wide area optical networks with large numbers of end nodes. We have investigated, applied and adapted some of the modern Operations Research (OR) techniques that are popular in the OR community for solving large mixed integer linear problems.

We have reported our works on three major aspects of optimal design of WDM optical network, such as:

- Optimal traffic grooming in WDM optical networks,
- Optimal regenerator placement in translucent networks, and
- Optimal lightpath allocation in translucent networks.

For optimal lightpath allocation in translucent optical networks, we have reported our three separate works, namely,

- Dynamic lightpath allocation in translucent networks,

- Dynamic survivable lightpath allocation in translucent networks, and
- Static lightpath allocation in translucent networks.

For optimal traffic grooming problem we have implemented branch, price and cut algorithm using arc-chain formulation. For optimal regenerator placement problem we have implemented branch and cut algorithm for large networks. For smaller networks we have presented a compact node-arc formulation to solve the problem. In our algorithms for dynamic lightpath allocation and dynamic survivable lightpath allocation, we have incorporate an important property for translucent networks that have been identified in [?] (discussed in section 5.1). Algorithms without incorporating this property might produce invalid solutions for the problem.

We have pointed out some special characteristics of static lightpath allocation problem in translucent networks. We have presented an ILP formulation, as well as, an efficient heuristic, to solve the problem.

With extensive simulation experiments, we have proved the effectiveness and efficiencies of all of our proposed algorithms.

## 6.2 Future Works

Until recently, only the linear impairments are considered, in terms of optical reach, during RWA in WDM networks. However, in practice the optical signals are subjected to non-linear effects also. A lightpath established only considering the linear impairments may not have acceptable BER values due to the nonlinear effects. Further, the BER value of a lightpath which was originally acceptable, may deteriorate to an unacceptable level after a new

lightpath has been established, due to non-linear effects of the new lightpath (e.g., inter-channel interferences and leakages in the optical switches).

Recently researchers are developing algorithms that consider the effects of non-linear physical impairments along with linear physical impairments during RWA in WDM optical networks. This type of approach is known as *physical layer impairment aware RWA* (PLIA-RWA) or simply *impairment aware RWA* (IA-RWA) [17]. However, when considering IA-RWA algorithms, researchers have suggested that it is useful to categorize the PLIs into those that affect a lightpath individually (Case 1) and those that are generated by the interference among lightpaths (Case 2) [17].

The interdependence between the physical and the network layers makes the RWA problem in the presence of impairments a cross-layer optimization problem [17]. An important distinction is how the IA-RWA algorithms define the interaction between the networking and the physical layers and if they jointly optimize the solutions over these two layers. Because of some particular interference-related impairments, routing decisions made for one lightpath affect and are affected by the decisions made for other lightpaths. This interference is particularly difficult to formulate in offline IA-RWA where there are no already established connections and the utilization of the lightpaths are the variables of the problem. It is because of this difficulty that the offline IA-RWA algorithms proposed to date do not handle interference-related impairments.

In the future, we want to implement the techniques we developed during my Ph.D. research, for optimal design of optical networks, such as, optimal IA-RWA, optimal IA-RPP and optimal IA-Traffic Grooming in WDM translucent optical networks.

# Copyright Releases

*1. A Branch, Price and Cut Approach for Optimal Traffic Grooming in WDM Optical Networks; Rahman, Q.; Bandyopadhyay, S.; Aneja, Y.; IEEE International Conference on Communications, June 2011, DOI: 10.1109/icc.2011.5962606.*

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line ©2011 IEEE.

2) In the case of illustrations or tabular material, we require that the copyright line ©[Year of original publication] IEEE appear prominently with each reprinted figure and/or table.

3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: ©[year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]

2) Only the accepted version of an IEEE copyrighted paper can be used

when posting the paper or your thesis on-line.

3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to [http://www.ieee.org/publications\\_standards/publications/rights/rights\\_link.html](http://www.ieee.org/publications_standards/publications/rights/rights_link.html) to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

***2. Dynamic Lightpath Allocation in Translucent WDM Optical Networks, Bandyopadhyay, Subir; Rahman, Quazi; Banerjee, Sujogya; Murthy, Sudheendra; Sen, Arunabha; IEEE International Conference on Communications, IEEE/ICC 2009, June 2009, Page(s): 1 - 6, DOI: 10.1109/ICC.2009.5198956.***

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line ©2011 IEEE.

2) In the case of illustrations or tabular material, we require that the copyright line ©[Year of original publication] IEEE appear prominently with each reprinted figure and/or table.

3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: ©[year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]

2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.

3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to [http://www.ieee.org/publications\\_standards/publications/rights/rights\\_link.html](http://www.ieee.org/publications_standards/publications/rights/rights_link.html) to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

**3. *Path Protection in Translucent WDM Optical Networks; Rahman, Q.; Bandyopadhyay, S.; Bari, A.; Jaekel, A.; Aneja, Y.; Distributed Computing and Networking, (LNCS), vol. 5935, 2010, DOI: 10.1007/978-3-642-11322-2\_38.***

This is a License Agreement between Quazi R Rahman ("You") and Springer ("Springer") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by Springer, and the payment terms and conditions.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

Terms and Conditions

Introduction

The publisher for this copyrighted material is Springer Science + Business Media. By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the Billing and Payment terms and conditions es-



License Number	2974961017805
License date	Aug 23, 2012
Licensed content publisher	Springer
Licensed content publication	Springer eBook
Licensed content title	Path Protection in Translucent WDM Optical Networks
Licensed content author	Q. Rahman
Licensed content date	Feb 9, 2010
Type of Use	Thesis/Dissertation
Portion	Full text
Number of copies	None
Author of this Springer article	Yes and you are the sole author of the new work
Order reference number	None
Title of your thesis/dissertation	Optimization of WDM Optical Networks
Expected completion date	Sep 2012
Estimated size(pages)	210
Total	0.00 CAD

tablished by Copyright Clearance Center, Inc. (“CCC”), at the time that you opened your Rightslink account and that are available at any time at <http://myaccount.copyright.com>).

#### Limited License

With reference to your request to reprint in your thesis material on which Springer Science and Business Media control the copyright, permission is granted, free of charge, for the use indicated in your enquiry.

Licenses are for one-time use only with a maximum distribution equal to the number that you identified in the licensing process.

This License includes use in an electronic form, provided its password protected or on the university’s intranet or repository, including UMI (according to the definition at the Sherpa website: <http://www.sherpa.ac.uk/romeo/>). For any other electronic use, please contact Springer at ([permissions.dordrecht@springer.com](mailto:permissions.dordrecht@springer.com) or [permissions.heidelberg@springer.com](mailto:permissions.heidelberg@springer.com)).

The material can only be used for the purpose of defending your thesis, and with a maximum of 100 extra copies in paper.

Although Springer holds copyright to the material and is entitled to negotiate on rights, this license is only valid, provided permission is also obtained from the (co) author (address is given with the article/chapter) and provided it concerns original material which does not carry references to other sources (if material in question appears with credit to another source, authorization from that source is required as well).

Permission free of charge on this occasion does not prejudice any rights we might have to charge for reproduction of our copyrighted material in the future.

#### Altering/Modifying Material: Not Permitted

You may not alter or modify the material in any manner. Abbreviations, additions, deletions and/or any other alterations shall be made only with prior written authorization of the author(s) and/or Springer Science + Business Media. (Please contact Springer at (permissions.dordrecht@springer.com or permissions.heidelberg@springer.com)

#### Reservation of Rights

Springer Science + Business Media reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.

#### Copyright Notice:Disclaimer

You must include the following copyright and permission notice in connection with any reproduction of the licensed material: "Springer and the original publisher /journal title, volume, year of publication, page, chapter/article title, name(s) of author(s), figure number(s), original copyright notice) is given to the publication in which the material was originally published, by adding; with kind permission from Springer Science and Business Media"

#### Warranties: None

Example 1: Springer Science + Business Media makes no representations or warranties with respect to the licensed material.

Example 2: Springer Science + Business Media makes no representations or warranties with respect to the licensed material and adopts on its own behalf the limitations and disclaimers established by CCC on its behalf in its Billing and Payment terms and conditions for this licensing transaction.

#### Indemnity

You hereby indemnify and agree to hold harmless Springer Science + Business Media and CCC, and their respective officers, directors, employees and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.

#### No Transfer of License

This license is personal to you and may not be sublicensed, assigned, or transferred by you to any other person without Springer Science + Business Media's written permission.

#### No Amendment Except in Writing

This license may not be amended except in a writing signed by both parties (or, in the case of Springer Science + Business Media, by CCC on Springer Science + Business Media's behalf).

#### Objection to Contrary Terms

Springer Science + Business Media hereby objects to any terms contained in any purchase order, acknowledgment, check endorsement or other writing prepared by you, which terms are inconsistent with these terms and conditions or CCC's Billing and Payment terms and conditions. These terms and conditions, together with CCC's Billing and Payment terms and conditions (which are incorporated herein), comprise the entire agreement between you and Springer Science + Business Media (and CCC) concerning this licensing transaction. In the event of any conflict between your obligations established by these terms and conditions and those established by CCC's Billing and Payment terms and conditions, these terms and conditions shall control.

#### Jurisdiction

All disputes that may arise in connection with this present License, or the breach thereof, shall be settled exclusively by arbitration, to be held in The Netherlands, in accordance with Dutch law, and to be conducted under the Rules of the 'Netherlands Arbitrage Instituut' (Netherlands Institute of Arbitration).OR:

All disputes that may arise in connection with this present License, or the breach thereof, shall be settled exclusively by arbitration, to be held in the Federal Republic of Germany, in accordance with German law.

Other terms and conditions:

v1.3

If you would like to pay for this license now, please remit this license along with your payment made payable to “COPYRIGHT CLEARANCE CENTER” otherwise you will be invoiced within 48 hours of the license date. Payment should be in the form of a check or money order referencing your account number and this invoice number RLNK500844399.

Once you receive your invoice for this order, you may pay your invoice by credit card. Please follow instructions provided at that time.

Make Payment To: Copyright Clearance Center Dept 001 P.O. Box 843006 Boston, MA 02284-3006

For suggestions or comments regarding this order, contact RightsLink Customer Support: [customercare@copyright.com](mailto:customercare@copyright.com) or +1-877-622-5543 (toll free in the US) or +1-978-646-2777.

Gratis licenses (referencing \$0 in the Total field) are free. Please retain this printable license for your reference. No payment is required.

**4. On Static RWA in Translucent Optical Networks; Rahman, Quazi; Bandyopadhyay, Subir; Aneja, Yash; IEEE Symposium on Computers and Communication (ISCC'12), July 2012, Page(s): 171 - 176, DOI: 10.1109/ISCC.2012.6249288.**

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line ©2011 IEEE.

2) In the case of illustrations or tabular material, we require that the copyright line ©[Year of original publication] IEEE appear prominently with each reprinted figure and/or table.

3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper

in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: ©[year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]

2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.

3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to [http://www.ieee.org/publications\\_standards/publications/rights/rights\\_link.html](http://www.ieee.org/publications_standards/publications/rights/rights_link.html) to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

# Bibliography

- [1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.
- [2] S. Azodolmolky, M. Klinkowskia, E. Marina, D. Careglio, J. S. Pareta, and I. Tomkos, “A survey on physical layer impairments aware routing and wavelength assignment algorithms in optical networks,” *Computer Networks*, vol. 53, no. 7, pp. 926 – 944, May 2009.
- [3] S. Bandyopadhyay, *Dissemination of Information in Optical Networks*. Springer, 2008.
- [4] S. Bandyopadhyay, Q. Rahman, S. Banerjee, S. Murthy, and A. Sen, “Dynamic lightpath allocation in translucent WDM optical networks,” in *IEEE International Conference on Communications, (IEEE/ICC)*, 2009, pp. 1–6.
- [5] A. Bari, Q. Rahman, A. Jaekel, and S. Bandyopadhyay, “Traffic grooming in WDM mesh networks with guaranteed survivability,” in *Dependable Systems and Networks (DSN) DCCS, 2008, 38th annual IEEE/IFIP International Conference on*, June 2008.

- [6] C. Barnhart, C. A. Hane, and P. H. Vance, “Using branch-and-price-and-cut to solve origin-destination integer multicommodity flow problems,” *Operations Research*, vol. 48, no. 2, pp. 318 – 326, March 2000.
- [7] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, “Branch-and-price: Column generation for solving huge integer programs,” *Operations Research*, vol. 46, pp. 316–329, 1996.
- [8] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear Programming and Network Flows*. Wiley, 1990.
- [9] “Branch and Bound Algorithm,” Information available online at: [http://en.wikipedia.org/wiki/Branch\\_and\\_bound](http://en.wikipedia.org/wiki/Branch_and_bound).
- [10] V. D. H. Boom, W. Li, V. P. K. Bennekom, I. T. Monroy, and G.-D. Khoe, “High-capacity transmission over polymer optical fiber,” *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 7, no. 3, pp. 461 – 470, May - June 2001.
- [11] C. Brackett, “Dense wavelength division multiplexing networks: Principles and applications,” in *IEEE Journal of Selected Areas in Communications*, August 1990, pp. 373 – 380.
- [12] T. Carpenter, D. Shallcross, J. Gannett, J. Jackel, and A. V. Lehmen, “Method and system for design and routing in transparent optical networks,” U.S. Patent 7,286,480 B2, October, 2007.
- [13] —, “Maximizing the transparency advantage in optical networks,” in *Optical Fiber Communication Conference*, vol. 2, no. 23-28, 2003, pp. 616–617.

- [14] B. Chatelain, S. Mannor, F. Gagnon, and D. V. Plant, “Non-cooperative design of translucent networks,” in *Global Telecommunications Conference (GLOBECOM '07)*, November 2007, pp. 2348 – 2352.
- [15] S. Chen, I. Ljubić, and S. Raghavan, “The regenerator location problem,” in *Networks*, vol. 55, no. 3, May 2010, pp. 205–220.
- [16] A. L. Chiu and E. Modiano, “Reducing electronic multiplexing costs in unidirectional sonet/wdm ring networks via efficient traffic grooming,” in *Global Telecommunications Conference, IEEE/GLOBECOM*, vol. 1, November 1998, pp. 322 – 327.
- [17] K. Christodoulopoulos, K. Manousakis, and E. Varvarigos, “Offline routing and wavelength assignment in transparent wdm networks,” *IEEE/ACM Transactions on Networking*, vol. 18, no. 5, p. 1557–1570, 2010.
- [18] V. Chvatal, *Linear Programming*. W. H. Freeman, 1983.
- [19] J. Clausen, “Branch and bound algorithms principles and examples,” *Parallel Computing in Optimization*, 1997.
- [20] F. P. P. da Cunha e Alvelos, “Branch-and-price and multicommodity flows,” Ph.D. Dissertation, Universidade do Minho, 2005.
- [21] G. B. Dantzig and P. Wolfe, “Decomposition principle for linear programs,” *Operations Research*, pp. 8101–8111, 1960.
- [22] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.



- [23] R. Dutta, S. Huang, and G. N. Rouskas, "On optimal traffic grooming in elemental network topologies," in *OptiComm 2003*, ser. 13 - 24, October 2003.
- [24] R. Dutta and G. N. Rouskas, "On optimal traffic grooming in WDM rings," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 1, pp. 110–121, January 2002.
- [25] "Erlang (unit)," Information available online at:  
[http://en.wikipedia.org/wiki/Erlang\\_\(unit\)](http://en.wikipedia.org/wiki/Erlang_(unit)).
- [26] J. Fang and A. K. Somani, "Enabling subwavelength level traffic grooming in survivable WDM optical network design," in *IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 5, December 2003, pp. 2761–2766.
- [27] A. L. S. Filho and H. Waldman, "Strategies for designing translucent wide-area networks," in *International Microwave and Optoelectronics Conference (IMOC)*, 2003, pp. 931–936.
- [28] M. Flammini, A. M. Spaccamela, G. Monaco, L. Moscardelli, and S. Zaks, "On the complexity of the regenerator placement problem in optical networks," in *IEEE/ACM Transactions on Networking*, 2010.
- [29] J. B. Georges, D. M. Cutrer, O. Solgaard, and K. Y. Lau, "Optical transmission of narrowband millimeter-wave signals," *IEEE Transactions on Microwave Theory and Techniques*, vol. 43, no. 9, pp. 2229 – 2240, September 1995.

- [30] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computing and Operations research*, vol. 13, no. 5, pp. 553 – 549, 1986.
- [31] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica (USA)*, vol. 20, no. 4, pp. 374–387, April 1998.
- [32] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," in *Systems Science and Cybernetics, IEEE Transactions on*, vol. 4, no. 2, 1968, pp. 100 – 107.
- [33] J. Q. Hu and B. Leida, "Traffic grooming, routing, and wavelength assignment in optical wdm mesh networks," in *IEEE INFOCOM*, March 2004, pp. 495 – 501.
- [34] J.-Q. Hu and E. Modiano, *Optical WDM Networks: Principles and Practice*. Kluwer Academic Publishers, 2004, vol. II, ch. Traffic Grooming in WDM Networks.
- [35] W. Huo, L. Guang, C. Assi, and A. Shami, "Survivable traffic grooming in optical networks with multiple failures," in *Canadian Conference on Electrical and Computer Engineering*, 2005.
- [36] "IBM ILOG CPLEX Optimizer," Documentation available online at: <http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/>.
- [37] A. Jaekel, A. Bari, and S. Bandyopadhyay, "Strategies for traffic grooming over logical topologies," in *Parallel and Distributed Processing and*

- Applications (LNCS)*. Springer Berlin / Heidelberg, 2007, vol. 4742, pp. 797–808.
- [38] —, “Resilient traffic grooming for WDM networks,” *Journal of Optical Networking*, vol. 7, no. 5, pp. 378–387, 2008.
- [39] T. R. Jensen and B. Toft, *Graph Coloring Problems*. Wiley, 1995.
- [40] C. Joergensen, T. Durhuus, C. Braagaard, B. Mikkelsen, and K. E. Stubkjaer, “4 gb/s optical wavelength conversion using semiconductor optical amplifiers,” in *IEEE Photonics Technology Letters*, vol. 5, no. 6, June 1993, pp. 657 – 660.
- [41] S. Kim, X. J. Zhang, and S. S. Lumetta, “Rapid and efficient protection for WDM mesh networks,” *IEEE Journal of Selected Areas in Communication*, 2007.
- [42] V. R. Konda and T. Y. Chow, “Algorithm for traffic grooming in optical networks to minimize the number of transceivers,” in *IEEE Workshop on High Performance Switching and Routing*, May 2001, pp. 218 – 221.
- [43] M. S. Kumar and P. S. Kumar, “Static lightpath establishment in WDM networks – New ILP formulations and heuristic algorithms,” *Computer Communications*, vol. 25, no. 1, pp. 109–114, January 2002.
- [44] L. Ladányi, T. K. Ralphs, and L. E. T. Jr., “Branch, cut, and price: Sequential and parallel,” in *Computational Combinatorial Optimization*, 2001, pp. 223–260.

- [45] L. Ladányi, T. K. Ralphs, and L. E. Trotter, “Branch, cut, and price: Sequential and parallel,” in *Computational Combinatorial Optimization*, D. Naddef and M. Jünger, Eds. Springer, Berlin, 2001, pp. 223–260.
- [46] S. Z. S. Lei, C. Vadrevu, and B. Mukherjee, “Network virtualization over wdm networks,” in *International Conference on Advanced Networks and Telecommunication Systems (ANTS)*, 2011, pp. 1 – 3.
- [47] A. Morea and I. B. Heard, “Availability of translucent networks based on WSS nodes, comparison with opaque networks,” in *Transparent Optical Networks, 2006 International Conference on*, vol. 4, June 2006, pp. 43 – 47.
- [48] A. Morea, H. Nakajima, L. Chacon, E. L. Rouzic, B. Decocq, and J.-P. Sebille, “Impact of the reach of WDM systems and traffic volume on the network resources and cost of translucent optical transport networks,” in *Transparent Optical Networks, 2004. Proceedings of 6th International Conference on*, vol. 1, July 2004, pp. 65 – 68.
- [49] A. Narula-Tam, E. Modiano, and A. Brzezinski, “Physical topology design for survivable routing of logical rings in WDM-based networks,” in *IEEE GLOBECOM Global Telecommunications Conference*, vol. 5, December 2003, pp. 2552 – 2557.
- [50] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*. John Wiley and Sons, 1988.
- [51] C. Ou, K. Zhu, H. Zang, L. H. Sahasrabudde, and B. Mukherjee, “Traffic grooming for survivable WDM networks — shared protection,” *IEEE*

- Journal on Selected Areas in Communications*, vol. 21, no. 9, pp. 1367–1383, November 2003.
- [52] A. E. Ozdaglar and D. P. Bertsekas, “Routing and wavelength assignment in optical networks,” in *IEEE/ACM Transactions on Networking*, vol. 11, no. 2, April 2003, pp. 259 – 272.
- [53] S. Pachnicke, T. Paschenda, and P. M. Krummrich, “Physical impairment based regenerator placement and routing in translucent optical networks,” in *Optical Fiber communication/National Fiber Optic Engineers Conference, OFC/NFOEC 2008, Conference on*, February 2008, pp. 1 – 3.
- [54] B. Ramamurthy, H. Feng, D. Datta, J. P. Heritage, and B. Mukherjee, “Transparent vs. opaque vs. translucent wavelength-routed optical networks,” in *Optical Fiber Communication Conference (OFC/IOOC), Technical Digest*, vol. 1, February 1999, pp. 59 – 61.
- [55] S. Ramamurthy and B. Mukherjee, “Survivable WDM mesh networks part I - protection,” in *IEEE International Conference on Computer and Communications (INFOCOM)*, vol. 2, March 1999, pp. 744–751.
- [56] —, “Survivable WDM mesh networks part II - restoration,” in *IEEE International Conference on Communications*, vol. 3, 1999, pp. 2023 – 2030.
- [57] R. Ramaswami and K. N. Sivarajan, “Design of logical topologies for wavelength-routed optical networks,” *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp. 840–851, June 1996.

- 
- [58] —, *Optical Networks: A Practical Perspective*. Morgan Kaufmann Publishers, 2002.
- [59] G. Rouskas and R. Dutta, *Optical WDM Networks: Principles and Practice*. Kluwer, 2000, ch. Design of Logical Topologies for Wavelength Routed Networks, pp. 79–102.
- [60] S. Rumley and C. Gaumier, “Cost aware design of translucent WDM transport networks,” in *International Conference on Transparent Optical Networks, ICTON '09*, June 2009, pp. 1 – 4.
- [61] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.
- [62] L. Sahasrabudde, S. Ramamurthy, and B. Mukherjee, “Fault management in IP-over-WDM networks: WDM protection versus IP restoration,” *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 1, pp. 21–33, January 2002.
- [63] N. Sambo, N. Andriolli, A. Giorgetti, F. Cugini, L. Valcarengi, and P. Castoldi, “Distributing shared regenerator information in gmpls-controlled translucent networks,” in *Communications Letters, IEEE*, vol. 12, no. 6, June 2008, pp. 462 – 464.
- [64] B. Sartorius, “3r regeneration for all-optical networks,” in *3rd International Conference on Transparent Optical Networks*, June 2001, pp. 333 – 337.

- [65] M. Scheffel, “Regenerator allocation strategies for optical transparency domains considering transmission limitations,” in *IEEE International Conference on Communications*, vol. 3, May 2005, pp. 1771 – 1776.
- [66] A. Sen, S. Banerjee, P. Ghosh, S. Murthy, and H. Ngo, “On regenerator placement and routing problems in optical networks,” in *ACM/SPAA*, June 2010.
- [67] A. Sen, S. Murthy, and S. Bandyopadhyay, “On sparse placement of regenerator nodes in translucent optical network,” in *Global Telecommunications Conference, (IEEE/GLOBECOM)*, 2008, pp. 1–6.
- [68] G. Shen, W. D. Grover, T. H. Cheng, and S. K. Bose, “Sparse placement of electronic switching nodes for low blocking in translucent optical networks,” *OSA Journal of Optical Networks*, vol. 1, 2002.
- [69] G. Shen, W. D. Grover, T. H. Cheng, and S. K. Bosh, “Sparse placement of electronic switching nodes for low blocking in translucent optical networks,” *OSA Journal of Optical Networking*, vol. 1, no. 12, December 2002.
- [70] G. Shen and R. S. Tucker, “Translucent optical networks: the way forward,” *IEEE Communications Magazine*, vol. 45, 2007.
- [71] N. Shinomiya, T. Hoshida, Y. Akiyama, H. Nakashima, and T. Terahara, “Hybrid link/path-based design for translucent photonic network dimensioning,” *Journal of Lightwave Technology*, vol. 5, no. 10, pp. 2931–2941, October 2007.

- [72] J. Simmons, “On determining the optimal optical reach for a long-haul network,” *Journal of Lightwave Technology*, vol. 23, March 2005.
- [73] —, “Network design in realistic all-optical backbone networks,” *IEEE Communications Magazine*, vol. 44, 2006.
- [74] A. K. Somani, *Survivability and Traffic Grooming in WDM Optical Networks*. Cambridge University Press, February 2006.
- [75] P. Soproni, P. Babarczi, J. Tapolcai, T. Cinkler, and H. Pin-Han, “A meta-heuristic approach for non-bifurcated dedicated protection in wdm optical networks,” in *International Workshop on the Design of Reliable Communication Networks (DRCN)*, 2011, pp. 110 – 117.
- [76] “Sun Fire X2200 M2 Server,” Documentation available online at: <http://www.sun.com/servers/x64/x2200/>.
- [77] “Fiber Optic Transceivers,” Information available online at: <http://www.thefoa.org/tech/ref/appln/transceiver.html>.
- [78] S. Thiagarajan and A. K. Somani, “A capacity correlation model for wdm networks with constrained grooming capabilities,” in *IEEE International Conference on Communications*, vol. 5, June 2001, pp. 1592 – 1596.
- [79] —, “Traffic grooming for survivable WDM mesh networks,” in *OptiComm 2001, Optical Networking and Communications*, vol. 4599, August 2001, pp. 54–65.
- [80] J. A. Tomlin, “Minimum-cost multicommodity network flows,” *Operations Research*, vol. 14, no. 1, pp. 45–51, January-February 1966.



- [81] R. ul Mustafa and A. E. Kamal, “Design and provisioning of WDM networks with multicast traffic grooming,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 4, pp. 37–53, April 2006.
- [82] L. A. Wolsey, *Integer Programming*. John Wiley and Sons, 1998.
- [83] B. Xiang, S. Wang, and L. Li, “A traffic grooming algorithm based on shared protection in WDM mesh networks,” in *Parallel and Distributed Computing, Applications and Technologies, 2003. Proceedings of the Fourth International Conference on*, August 2003, pp. 254 – 258.
- [84] C. Xin, B. Wang, X. Cao, and J. Li, “Logical topology design for dynamic traffic grooming in wdm optical networks,” *Journal of Lightwave Technology*, vol. 24, no. 6, pp. 2267 – 2275, June 2006.
- [85] X. Yang and B. Ramamurthy, “Dynamic routing in translucent WDM optical networks: The intradomain case,” *Journal of Lightwave Technology*, vol. 23, no. 3, pp. 955–971, March 2005.
- [86] —, “Sparse regeneration in translucent wavelength routed optical networks: architecture, network design and wavelength routing,” *Photonic network communications*, vol. 10, 2005.
- [87] —, “Dynamic routing in translucent WDM optical networks: the interdomain case,” *Journal of Lightwave Technology*, vol. 23, March 2005.
- [88] X. Yang, L. Shen, and B. Ramamurthy, “Survivable lightpath provisioning in WDM mesh networks under shared path protection and signal quality constraints,” *Lightwave Technology, Journal of*, vol. 23, no. 4, pp. 1556 – 1567, April 2005.

- [89] W. Yao and B. Ramamurthy, “Survivable traffic grooming with path protection at the connection level in WDM mesh networks,” in *International Conference on Broadband Networks (BroadNets)*, October 2004, pp. 310–319.
- [90] Y. Ye, T. H. Cheng, and C. Lu, “Routing and wavelength assignment algorithms for translucent optical networks,” *Optics Communications*, vol. 229, no. 1 - 6, pp. 233 – 239, January 2004.
- [91] H. Zang, J. P. Jue, and B. Mukherjee, “A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks,” *SPIE Opt. Net. Mag.*, vol. 1, no. 1, January 2000.
- [92] J. Zhang and B. Mukherjee, “A review of fault management in WDM mesh networks: Basic concepts and research challenges,” *IEEE Network*, vol. 18, no. 2, pp. 41–48, March-April 2004.
- [93] J. Zheng and H. T. Mouftah, *Optical WDM Networks: Concepts and Design Principles*. John Wiley & Sons, 2004.
- [94] Y. Zheng, W. Gu, S. Huang, and P. Zhang, “An ant-based research on rwa in optical networks,” in *International Conference on Electronic Computer Technology*, 2009, pp. 73 – 76.
- [95] B. Zhou, S. R. Pramod, and H. T. Mouftah, “Adaptive ber-assured routing in translucent optical networks,” in *Workshop on High Performance Switching and Routing*, 2004, pp. 209–213.
- [96] D. Zhou and S. Subramaniam, “Survivability in optical networks,” in *IEEE Network*, Nov-Dec 2000, pp. 16 – 23.

- 
- [97] K. Zhu and B. Mukherjee, “Traffic grooming in an optical WDM mesh network,” *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 1, pp. 122–133, January 2002.
- [98] —, “A review of traffic grooming in WDM optical networks: Architectures and challenges,” *Optical Networks Magazine*, vol. 4, no. 2, pp. 55–64, March 2003.
- [99] K. Zhu, H. Zang, and B. Mukherjee, “A comprehensive study on next-generation optical grooming switches,” *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 7, pp. 1173–1186, September 2003.
- [100] K. Zhu, H. Zhu, and B. Mukherjee, *Traffic Grooming in Optical WDM Mesh Networks*. Springer, 2005.

# Vita Auctoris

Quazi Reshadur Rahman was born in 1955 in Joypurhat, Bangladesh. He passed the Secondary School Certificate examination in 1970 from Kalai High School, Joypurhat, Bangladesh. In 1972, he passed the Higher Secondary Certificate examination from Azizul Haque College, Bogra, Bangladesh. From there he went to Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, where he obtained B.Sc. in Electrical Engineering degree in 1978. He obtained M.S. in Computer Science from The City College of New York, USA in 1997. He obtained M.Sc. in Computer Science degree from the University of Windsor, Ontario in 2008. He is currently a candidate for the Doctoral degree in Computer Science at the University of Windsor, Ontario and hopes to graduate in Summer 2012.