

“The most successful men in the end are those whose success is the result of steady accretion... It is the man who carefully advances step by step, with his mind becoming wider and wider—and progressively better able to grasp any theme or situation—persevering in what he knows to be practical, and concentrating his thought upon it, who is bound to succeed in the greatest degree.”

A Quote by **Alexander Graham Bell (1847-1922)**
on men, mind, practicality, success, and thought.

University of Alberta

***p*-Cycles: New Solutions for Node Protection, Transparency,
and Large Scale Network Design**

by

Diane Prisca Onguetou Boaye

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

in

Communications

Department of Electrical and Computer Engineering

© Diane Prisca Onguetou

Fall 2011

Edmonton, Alberta

Permission is hereby granted to the University of Alberta Libraries to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only. Where the thesis is converted to, or otherwise made available in digital form, the University of Alberta will advise potential users of the thesis of these terms.

The author reserves all other publication and other rights in association with the copyright in the thesis and, except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatsoever without the author's prior written permission.



Examining Committee

Dr. Wayne D. Grover, Electrical and Computer Engineering, University of Alberta

Dr. Bruce Cockburn, Electrical and Computer Engineering, University of Alberta

Dr. Ivan Fair, Electrical and Computer Engineering, University of Alberta

Dr. Ehab Elmallah, Computing Science, University of Alberta

Dr. James Sterbenz, Electrical Engineering and Computer Science, Information and
Telecommunication Technology Center, University of Kansas

“To Christ Jesus our Lord, who has strengthened me, 1 Timothy 1 : 12.”



Abstract

Optical transport networks are critical infrastructures which carry huge amounts of great traffic variety but are inevitably subject to laser diode failures, fiber cuts and sometimes node outages. The concept of p -cycles is very attractive and competitive in the domain of network survivability because p -cycles have a unique ability to combine the real-time switching simplicity and speed of rings with the capacity efficiency, flexibility and freedom of a mesh in the routing of working and restored state paths.

This dissertation presents several new research studies that increase our knowledge and act of available techniques to use and understand p -cycles. Advancements include a relatively simple but cost-effective generalization of how a BLSR-ring (or p -cycle to-date) derives survivability, in the event of node failure, through loopback at the nearest two neighbor-nodes on the same cycle. Significantly, this new insight also gives rise to a novel two-hop segment protection paradigm that unifies node and span failure protection.

As well, the thesis introduces two fundamental advances for dealing with optical network transparency. One is the complementary matching of longer working paths with shorter protection segments available through p -cycles, thereby controlling the optical reach in restored network states. The other is the in-depth consideration of glass-switched p -cycles to rapidly, simply and efficiently provide for the direct replacement of failed fiber sections with whole replacement fibers. Experiments highlight that p -cycles formed out the span fibers overcome the complexity due to wavelength continuity requirements in transparent-

based designs, significantly reduce overall capital expenditure (CapEx) costs and provide a solid working capacity envelope for dynamic traffic considerations.

We also make advances on the problem of solving very large scale p -cycle design problems, with a technique that combines Genetic Algorithms (GA) with Integer Linear Programming (ILP). Basically, the GA-ILP considers any p -cycle ILP (to be solved) as the fitness function for a GA-like evolutionary heuristic, aimed at preselecting a few manageable candidate cycles working well together, from an almost infinite space. Beside the GA-ILP conceptual simplicity, experiments show high quality design solutions to very large scale p -cycle problem instances, involving up to 200 nodes.

Acknowledgement

“Great discoveries and improvements invariably involve the cooperation of many minds. I may be given credit for having blazed the trail but when I look at the prior and subsequent developments I feel the credit is due to others rather than to myself,”

Alexander Graham Bell (1847 - 1922),
a quote on cooperation and discovery.

At the end of this doctorate program, I would like to express my sincere gratitude to all those who in some way contributed to the completion of my PhD, as well as the University of Alberta (UofA), Telecommunications Research Laboratories (TRLabs) and Nokia Siemens Networks (NSN) who provided the working environment/material and grants.

I am especially grateful to my supervisor Dr. Wayne D. Grover, who gave me the opportunity to pursue my doctoral studies at UofA and TRLabs, and enrolled me in the HAVANA collaborative work with NSN. Despite the language barrier when I approached him a few years ago, he recognized my ability to conduct a successful research, supported my affiliation with TRLabs, and provided the financial support almost to the end. Dr. Grover did not only agree to supervise my PhD journey, but he also pushed me to give the best of myself and to grow up as a researcher. I was impressed by his unique ability to generate and conduct new research ideas/projects. And I hope this dissertation and related publications meet his expectations and are up to his investment of time and grants.

Beside Dr. Grover, I would like to mention Dr. Bruce Cockburn and Dr. Ivan Fair who also served as members of my supervisory committee, as well as external examiners Dr. Ehab Elmallah from the computing science center and Dr. James Sterbenz from the University of Kansas. I thank them all for their valuable inputs in this dissertation and hope

the final version reflect the revisions prescribed during my PhD defense. And I would also like to thank Dr. Dominic Schupke and Dr. Matthieu Clouqueur from NSN for the 3-year HAVANA project and related funds. This gave me some indirect industry experience, as well as the opportunity to bridge academic research and industry goals through real-world applications of fundamental research questions; many topics addressed within this thesis were generated and/or extended from the HAVANA research mandates. To all my former colleagues from TRILabs, especially Dimitri Baloukov, Brian Forst and Dr. Aden Grue from the Network Systems group, I really enjoyed our meetings and collaborative work for the HAVANA project. Those scientific discussions and tips truly enhanced my research, and they also helped me to mature as a teamwork player. And to Linda Richens and Rhoda Hayes from TRILabs, I appreciated the hand given for using facilities and process administrative tasks, as well as our talks that made TRILabs a warm place.

I cannot forget my dear parents and siblings who followed me from the beginning to the end of my PhD program, and encouraged me when I felt discouraged. To my parents Theophile and Celestine Onguetou in Yaoundé (Cameroon), I feel blessed to have you as parents. Be proud of raising your children in the fear of God, and instilling them the value of advanced education and job well done; thanks a lot for your prayers and advices. To my siblings and their relatives Radar Jones in the US, Esther Julie (her husband and their kids Nathan, Celeste and Jacquin) in Garoua (Cameroon), Jeannot-J and the benjamin Paule Jeanine in Yaoundé (Cameroon), I say thank you for loving and trusting me; a special mention to Paule for our daily chat on Internet, who used to make sure I was doing well. And to Michele Patricia in Montréal (Canada), I pray God to make a way for you.

Also to my friends around the world, we will never count on hours spent online or on the phone, sharing our life experiences and advices & tips, talking about everything and nothing, etc. I almost forgot how far was my family, as you guys made a perfect substitute. Particularly to Serge Christian Afana, my special friend met almost two years ago, you were just in time to help overcome the shortcomings and share the joy of this milestone in my life.



Contents

1	Introduction and Outline	1
1.1	Causes of Network Failures and Impact	2
1.2	Survivability in the Transport Layer	3
1.3	The Case for p -Cycles	5
1.4	Research Objectives and Highlights	6
2	Background on IP-over-WDM and p-Cycle Survivable Networks	10
2.1	Introduction to Transport Networking	10
2.1.1	Layered Architectural Model	12
2.1.2	Towards IP-over-WDM Networks	14
2.1.3	Synchronous Optical Network (SONET) Standard	15
2.1.4	Dense Wavelength Division Multiplexing (DWDM)	17
2.1.5	Generalized Multi-Protocol Label Switching (GMPLS)	20
2.1.6	The Next Step in the Optical Network Evolution: OTN ¹	22
2.2	Survivable Transport Networks	31
2.2.1	Fundamentals for Survivability in the Transport Layer	31
2.2.2	Transmission System Layer Survivability Schemes	35
2.2.3	Logical Layer Mesh-Type Survivability Schemes	39
2.2.4	Pre-cross-connected Protection Structures	45
2.3	Pre-configured Pre-cross-connected Protection Cycles	48
2.3.1	Basic Operating Principle, Ring and Mesh-like Benefits	48
2.3.2	Whole p -Cycle Network Design	50
2.3.3	Configuration in IP-over-WDM Networks	55
2.4	p -Cycle Literature Survey	58
2.4.1	Implementation and Operation in IP-over-WDM Networks	59

2.4.2	NEPC, Flow and FIPP Extended Protection Principles	60
2.4.3	Dual Failures, Optical Reach Control and Demand Uncertainty	60
2.4.4	Questions Revisited and Research Methodology	61
3	Node-Protecting p-Cycles	71
3.1	The Question of Node-Protecting p -Cycles	72
3.1.1	Node-Encircling p -Cycles (NEPCs)	73
3.1.2	Path-Segment Protecting p -Cycles	73
3.1.3	Failure-Independent Path-Protecting (FIPP) p -Cycles	74
3.2	A Simple Generalized Approach to Node Failure Recovery with Span-Protecting p -Cycles	75
3.2.1	New Insights and Principle	76
3.2.2	ILP Mathematical Formulations	77
3.2.3	Experimental Results and Discussion	81
3.3	A Two-Hop Segment Protecting Paradigm which Unifies Node and Span Failure Recovery under p -Cycles	86
3.3.1	Two-Hop Protecting p -Cycles	87
3.3.2	An ILP Design Model for Two-Hop Protecting p -Cycles	88
3.3.3	Two-Hop Protecting p -Cycles vis-à-vis Other p -Cycle Schemes	90
3.3.4	Summary of Two-Hop Protecting p -Cycle Features	94
3.4	Near-Optimal FIPP Network Designs using General Path-Protecting p -Cycles	95
3.4.1	Current FIPP Design Methods	95
3.4.2	General Path-Protecting (GPP) p -Cycles	97
3.4.3	Deriving a FIPP Solution from an Existing GPP Design	99
3.4.4	Effectiveness of FIPP Design Solutions through GPPs	100
3.4.5	Node Failure Protection using FIPPs through GPPs	101
3.5	Concluding Discussion	102
4	Fully Transparent p-Cycle Designs	104
4.1	p -Cycle Design with Controlled Optical Path Lengths in the Restored Net- work State	105
4.1.1	Prior Related Work	106

4.1.2	Matching Longer Working Paths with Shorter Path-Segments in the Available p -Cycles	107
4.1.3	ILP Models for Complementary Matching of Working Paths and Protection Segments	110
4.1.4	Case Studies and Experimental Results	115
4.1.5	Which is Preferable—Lower Average Path Length, Shortest Minimax or Fixed Optical Reach Limit?	120
4.2	p -Cycle Protection at the Glass Fiber Level	121
4.2.1	Chronological History of Glass Switched p -Cycles and Objectives . .	122
4.2.2	Whole Fiber Cross-Connect Switches	124
4.2.3	Complexity Reduction of Fully Transparent p -Cycle Design Models .	126
4.2.4	CapEx Cost Enhancements	129
4.3	Closing Discussion	140
5	A GA-ILP Heuristic for p-Cycle Design on Very Large Scales	142
5.1	The Large Scale p -Cycle Design Problem and Solutions	142
5.1.1	Preselection Methods	143
5.1.2	Unifying Enumeration-free ILPs within the Transportation-Flow Problem Structure	146
5.1.3	The Column Generation Approach	148
5.2	Towards an Exclusive Combination of GA-Methods and ILP to Solve the Large Scale p -Cycle Design Problem	149
5.2.1	General Understanding of the GA-ILP Philosophy	149
5.2.2	GA-ILP Evolutionary Steps and Programming Aspects	151
5.2.3	Effectiveness and Performance of the GA-ILP	155
5.2.4	GA-ILP Solution Quality vis-à-vis that of Other Practical Methods for Solving p -Cycle Design Problems	160
5.2.5	Run Time Issues	160
5.3	A 200-node Challenge Instance of the Conventional p -Cycle Minimum Spare Capacity Design Problem	162
5.3.1	The 200-node Test Case Network	162
5.3.2	Facing Infinite Candidates within the GA-ILP Framework	163

5.3.3	GA-ILP Subsidiary versus Final p -Cycle Design Models	164
5.3.4	Detailed Analysis of the DFS Local-Cycle Enumerator	166
5.3.5	GA-ILP Convergence under a Surrogate p -Cycle ILP	167
5.3.6	Run Times	170
5.4	p -Cycle Design Problems with Many Complicated Practical Constraints . .	170
5.4.1	Truth about Sample Results in Previous Chapters	171
5.4.2	Effectiveness of the GA-ILP for Solving Advanced p -Cycle Network Design Problems	171
5.4.3	Multi-Criteria Optimization	176
5.5	Concluding Points and Possible Future Directions	177
6	Conclusion and Further Work	179
6.1	Summary of the Contribution	179
6.1.1	Node-Protecting p -Cycles	179
6.1.2	CapEx Concerns and Transparency Fundamentals	181
6.1.3	p -Cycle Design on a Very Large Scale	182
6.2	Innovative Aspects of the Thesis and Implications	184
6.3	Possible Directions for Future Work	186
6.3.1	GA-ILP Improvements	186
6.3.2	Multilayer Network Planning	187
6.3.3	Hybrid Architecture Designs	187
6.3.4	Specific Ideas for p -Cycle Architecture Enhancements	188
6.4	Publications Associated with Thesis' Research Studies	188
6.4.1	Book in Preparation	188
6.4.2	Provisional Patent Applications	189
6.4.3	Journal Publications	189
6.4.4	Peer-Reviewed Conference Papers	189
6.4.5	Additional Presentations and Technical Reports	191
	Bibliography	192
	A Portrayal of a Conventional p-Cycle Design	206

B ILP Design Models for Wavelength Switched p-Cycles	209
B.1 Symbology	209
B.2 ILP for Opaque Wavelength Switched p -Cycles	210
B.3 ILP Formulation for Hybrid p -Cycle Designs	211
B.4 ILP for Fully Transparent p -Cycle Designs	212



List of Tables

- 2.1 Characteristics of Test Case Network Instances from the ILP₁ Perspective . . . 64

- 3.1 ILP Mathematical Formulations for Node-Protecting p -Cycles 77
- 3.2 Characteristics of Test Case Networks from the Node Failure Perspective . . 82
- 3.3 Sample Results for Current Node Recovery Options using p -Cycles 84
- 3.4 Sample Results for Comparative Study of p -Cycle Related Architectures . . 91
- 3.5 Benefits of Two-Hop Protecting p -Cycles over of Other Related Schemes . . 94
- 3.6 Sample Results for FIPP Design Solutions through GPP 101

- 4.1 Summary of the ILP Mathematical Models Introduced in this Chapter . . . 110
- 4.2 Sample Results for Restored State Path Length Minimization in the Cost239 116
- 4.3 Variables and Constraints Involved in Wavelength and Glass Switched p -
Cycle ILPs 128
- 4.4 Summary of Equipment Pricing Under the NOBEL Cost Model 134
- 4.5 Sample Results for Overall Real CapEx Costs 136

- 5.1 The GA-ILP vis-à-vis Prior Approaches: Sample Results for Euronet and
Cselt 160
- 5.2 Deeper Analysis of GA-ILP Run Time Trends, for Euronet and Cselt 161
- 5.3 Characteristics of the 200-Node p -Cycle Network Design Solution 170
- 5.4 Complexity of Advanced p -Cycle ILPs for Several Network Instances 172
- 5.5 Running the GA-ILP₈ with Different Population Sizes for the Italy Network 175



List of Figures

- 2.1 Segmentation of Telecommunication Networks in Multiple Geographical Tiers, adapted from [Gro02a, Gro03b, Dou06]. 11
- 2.2 Aggregation of Service Layer Traffic into Transport Demands, adapted from [Gro02a, Gro03b]. 12
- 2.3 Layered Architectural Model, adapted from [Gro02a, Gro03b]. 13
- 2.4 Functional Block Diagram of an OXC, adapted from [Dou06, Gro03b]. 19
- 2.5 OTN: an Evolution Platform for Multiple Services, [BJ10]. 23
- 2.6 Similarities and Improvements to SONET/SDH, [BJ10]. 25
- 2.7 Optical Transport Module, [BJ10]. 26
- 2.8 OTN Management Supervision, [BJ10]. 28
- 2.9 OTN Multiplexing Structure, [BJ10]. 29
- 2.10 OTN Frame Structure and Bitrates, [BJ10]. 30
- 2.11 Reed-Solomon FEC RS (255/239), [BJ10]. 31
- 2.12 Disjoint and Distinct Routes, adapted from [Gro02a, Gro03b]. 33
- 2.13 Not-connected, Connected, Two-connected and Bi-connected Graphs, adapted from [Gro02a, Gro03b]. 33
- 2.14 Operational Principle of UPSR, adapted from [Gro02a, Gro03b]. 37
- 2.15 Operational Principle of BLSR and Illustration for Possible Bandwidth Re-use, adapted from [Gro02a, Gro03b]. 38
- 2.16 Illustrating Shared Backup Path Protection, with solid lines representing working paths while colour-matching dashed lines are for replacement paths, adapted from [Gro02a, Gro03b]. 42
- 2.17 Illustrating Path Restoration, adapted from [Dou06]. 43
- 2.18 Illustrating Span Protection/Restoration, adapted from [Dou06]. 44

LIST OF FIGURES

2.19	Operating Principle of Span and Path-Protecting p -Trees (in blue), adapted from [GFO ⁺ 06].	46
2.20	Illustrating PXT Operating Principle, adapted from [Gro03b, Gro02a, GFO ⁺ 06].	47
2.21	Operating Principle of Span-Protecting p -Cycles, adapted from [Gro03b, Gro02a].	49
2.22	Impact of Straddling Failure Restoration on the Redundancy, adapted from [Gro03b, Gro02a].	50
2.23	An Example of a Whole p -Cycle Design.	51
2.24	ADM-like Nodal Device for p -Cycle-based Networking, adapted from [Gro03b, Gro02a].	56
2.25	Specific Requirements for the Logical Layer Configuration of p -Cycles. . . .	57
2.26	Operation of IP-Layer p -Cycles, adapted from [Gro03b, Gro02a].	59
2.27	Havana Test Case Instances.	65
2.28	Cost239 Test Case Network.	66
2.29	Italy Test Case Network.	66
2.30	Bellcore Test Case Network.	67
2.31	Euro Test Case Network.	68
2.32	Euronet Test Case Network.	69
2.33	Cselt Test Case Network.	69
2.34	The 200-node Challenge Topology: 200 nodes, 394 spans; working capacities atop spans.	70
3.1	Illustration of the NEPC Concept, adapted from [Gro03b, Dou06].	73
3.2	Flow-Protecting p -Cycles X and Y Handling Path-Segments]A,C[and]B,D[.	74
3.3	Intrinsic “Two-hop” Node-Protecting Capabilities of p -Cycles.	75
3.4	Havana Network Characterization from the Node Failure Perspective. . . .	83
3.5	Two-Hop Segments: Graph Transformation and Routing.	87
3.6	Two-Hop Segment Protection using p -Cycles.	88
3.7	Impact of Path Hop-counts on the Performance of Two-Hop Protecting p -Cycles.	92
3.8	Impact of Average Nodal Degree on the Performance of Two-Hop Protecting p -Cycles.	93

LIST OF FIGURES

3.9	FIPP Configuration Types: Type ₁ -condition (i); Type ₂ -both conditions (i) and (ii).	95
3.10	Illustration of the GPP Concept.	97
4.1	Matching Long Working Paths with Short Path-Segments in the Available p -Cycles.	108
4.2	Optical Transmission Length of Working Paths (alone) in the COST239 Network.	117
4.3	Effects of Asserting a Direct Optical Reach Limit on Spare Capacity Requirements	120
4.4	p -Cycle Configuration Types in the WDM Layer.	123
4.5	Protection Switching Adaptations of the NOBEL Cost Model, adapted from [Nob].	129
4.6	Example of Transparent Optical Path under the NOBEL Cost Model for WDM Layers, adapted from [Nob].	131
4.7	A Baseline Node Architecture under the NOBEL Cost Model for WDM Layers, adapted from [Nob].	131
5.1	The GA-ILP Preselection Heuristic	151
5.2	Large Scale Network Instances	156
5.3	Convergence of the GA-ILP for a Given Population Size.	157
5.4	Influence of PopulationSize on the GA-ILP Performance.	158
5.5	GA-ILP Reduced-Sets of Candidates.	159
5.6	The 200-node Challenge Topology: 200 nodes, 394 spans and a cosmologically large scale of candidate structures.	162
5.7	DFS Local-Cycle Enumeration	166
5.8	GA-ILP Convergence for the Fourth Node Inspection.	168
5.9	Development of the Fully Span Restorable 200-Node Network Design.	169
5.10	GA-ILP _{5 and 8} for full R _{1-node} and General Path-Protecting p -Cycles.	174
A.1	Network Solution and p -Cycle Selection	206
A.2	Proof of Effectiveness	208



List of Abbreviations

[A]

AC Ant colony

ADM Add/drop multiplexer

AIS Alarm inhibit signal

AMPL A modeling language for mathematical programming

APS Automatic protection switching

ART Telecommunications Regulation Agency (France)

ATM Asynchronous transfer mode

[B]

B-DCS Broadband DCS

BER Bit error rate

BLSR Bidirectional line-switched rings

[C]

CapEx Capital expenditure

CG Column generation

CIDA Capacitated iterative design algorithm

CO Telecommunication central office

CPLEX Simplex method and C programming language in the ILOG optimization software package

[D]

DCF Dispersion-compensating fiber

DCPC Distributed cycle pre-configuration protocol

DCS Digital cross-connect

DFS Depth first search algorithm

DGE Dynamic gain equalizer

DP APS Diverse path APS

DRS Disjoint route set

DS- x Digital signal of $x \cdot 64$ kbps

DSP Demand-wise shared protection

DWDM Dense-WDM

[E]

EXC Electrical cross-connect

[F]

FCC Federal Communications Commission

FEC Forwarding equivalence class OR forward error correction, depending on the context

FIPP Failure-independent path-protecting p -cycles

[G]

GA Genetic algorithms

GA-ILP Novel combination of GA-based methods and ILP

GbE Gigabit Ethernet

GFP Generic framing protocol

GMPLS Generalized multi-protocol label switching

GPP General path-protecting p -cycles

[H]

HAVANA High availability network architectures, collaborative work with Nokia-Siemens.

Havana Master network used in the HAVANA protect

[I]

IA Inline amplifier

ID Identification

IEEE Institute of electrical and electronics engineers

ILOG

ILP Integer linear programming

IP Internet Protocol

ISP Internet service provider

ITU International telecommunications union

ITU-T ITU standardization telecommunications sector

[J]

JCP Joint capacity placement, for simultaneous working and spare capacity optimization

[L]

LAN Local area network

LDP Label distribution protocol

LER MPLS-based label edge router

LIB Label information base

LP Linear programming

LSP MPLS-based label switched path

LSR MPLS-based label switching router

[M]

MCMF Multi-commodity maximum flow

MEMS Micro-electro-mechanical system

MIPGAP Mixed integer linear programming gap of optimality

MPLS Multi-protocol label switching

MTD Maximum transmission distance

Multi-QoP Multiple quality-of-protection

MUPS Maximum unit path straddlers

[N]

NEPC Node-encircling *p*-cycle

NGI Next generation Internet

NISS Node-inclusive span survivability

NOBEL Next generation optical network for broadband European leadership

[O]

OADM Optical add/drop multiplexer

OAM Operation administration and maintenance

OCh OTN Optical channel

OC-n SONET Optical carrier signal-level

ODU OTN OCh data unit

O-E-O Electro-optical conversion

OMS OTN Optical multiplex section

OpEx Operating/revenue expenditure

OPU OTN OCh payload unit

OSC Optical supervisory channel

OTM OTN Optical transport module

OTN (New generation) optical transport network

OTS OTN Optical transmission section

OTU OTN OCh transport unit

OXC Optical cross-connect

[P]

p-Cycle Pre-configured, pre-connected, protection cycle

PWCE Protected working capacity envelope

PXT Pre-cross-connected trails

[Q]

QoP Quality of protection

QoS Quality of service

[R]

R_{1-node} Single node failure restorability

R_{1-span} Single span failure restorability

R_{2-span} Dual span failure restorability

RoS Reliability of service

RS FEC Reed-Solomon code

[S]

SA Simulated annealing

SBPP Shared backup path protection

SCMF Single-commodity maximum flow

SCP Spare capacity placement optimization

SDH Synchronous digital hierarchy, SONET equivalent elsewhere (as opposed to North America), ITU-T G.957

SLA Straddling link algorithm

SNR Signal-to-noise-ratio

SONET Synchronous optical network, North America ANSI T1.105

SPE SONET synchronous payload envelope

SRLG Shared risk link group

STM-n Synchronous transport module

STS-n Synchronous transport signal-level

STS-Nc Synchronous transport signal-level for a clear-channel

[T]

TCM OTN Tandem connection monitoring

TDM Time division multiplexing

TS Tabu search

[U]

ULH Ultra long haul

UPSR Unidirectional path-switched rings

[V]

VC SONET virtual concatenation

VP ATM virtual path

VT SONET virtual tributary

[W]

W-DCS Wideband DCS

WDM Wavelength division multiplexing

Chapter 1

Introduction and Outline

Modern-day living heavily depends on the availability and reliability of telecommunication networks. As such basic facilities as water, power supplies, public health, transportation and buildings which enable the society to function, telecommunications are essential to an industrialized economy, lifestyle conditions, finance, education, entertainment and so on. To lay people, the best known telecommunication infrastructures pertain to access technologies which directly relate to the consumer—e.g. cell and residential telephone, Internet, television, banking machines. But in practice, all of those access technology systems rely on a single *transport network* aimed at trunking between carrier central offices, cell-phone companies' base stations and multiple transport backbones. A subsequent simplistic view of the whole network architecture is a *service/client layer* that manages customer data packets coming from different access or other client networks, and a *transport layer* which offers facilities and pieces of equipment to reliably deliver customer data.

Wondering which physical medium is well-suited for transport backbones, [EGRG05] describes *fiber-optic based transport networks* as one of the "engineering marvels of the 20th century, now a fundamental infrastructure, crucial to current and future economies and societies." Optical transport networking uses dense wavelength division multiplexed (DWDM) transmission technology and optical cross-connects (OXCs) which allow, nowadays, up to 100 Tbps of voice, video and (mainly) data flowing through a single fiber optic. Based on the state-of-the-art SONET which reports 120,000 voice conversations riding onto a single pair of fibers for transmission systems operating at 10 Gbps, the 100 Tbps of DWDM are equivalent to as many as 1.2 billion simultaneous voice calls on a single fiber optic [Mor01]. With such huge amounts of traffic variety, optical transport networks using DWDM technology constitute a source of substantial advantages for society, industry and the economy.

1.1 Causes of Network Failures and Impact

Ironically, a side effect of fiber optic and DWDM's enormous advantages are major and unexpectedly severe economic, personal and societal impacts that will inevitably arise if the optical backbone is subject to failures, even if it is just a temporary outage. For example, [Gro03b] quotes statistics citing direct voice-calling revenue losses of \$100,000 per minute in the event of major trunk group failure. On the basis of current reports of bankruptcies from an hour or more of outages, he suspects an increasing exposure of growing e-commerce web transactions. Even for large US companies, he quotes the **Gartner research group** which attributes up to \$500 millions in losses to network failures affecting critical business functions over a year alone [Gar].

To avoid revenue loss resulting from business disruption, many "mission critical" businesses must be available over communication networks 24 hours a day, seven days a week. But the information society is based on a surprisingly vulnerable fiber optic medium in which failures arise surprisingly frequently. [Cra93] classifies immediate causes of fiber optic breakdowns, in a comprehensive survey on the frequency and causes of fiber optic cable failures, as dig-ups at 58.1%, vehicles at 7.5%, human errors at 6.9%, power lines at 4.4%, rodents at 3.8%, sabotage at 2.5%, fire at 1.9%, firearms, flood, excavation and tree falls at 1.3% each, all of the four totaling 10%. He estimates the physical repair time at 14 hours on average, with a very high variance reaching up to 100 hours, and the mean time of service outages at about 5.2 hours. These numbers pose significant problems for such essential services as 911, travel booking, education, financial or the stock market.

Why not just bury the cables suitably deep or put them in conduits and stress that everyone should be careful when digging? [Gro03b] points out that what seems so simple is actually not in practice. He says, it does not matter how advanced the fiber optic technology is, *it is in a cable*; so even with the best physical protection measures, it will be damaged and with surprising frequency. To corroborate, [TN94] estimates the lifetime of a fiber optic cable mile at about 228 years; although this sounds reassuring, it is equivalent to 4.39 cuts per year per thousand sheath-miles, which implies more than one cut per day on 100,000 installed route miles. In the same vein, [JP02] quotes FCC statistics of 13 cuts for every 1000 miles fiber per year in metro networks, which means even the lower rate

for long haul experiences a cable cut every four days in an atypical network with 30,000 route-miles of fiber.

To conclude, [EGRG05] states that “cable-cutting events virtually occur every few days in extensive networks with 50,000 or more route-miles of fibers. To the extent that construction activities correlate with the working week, fiber optic cable failures may also tend to cluster, producing days in which perhaps two or three cuts occur.” For him, “every effort can be made to protect the relatively few thumb-sized cables on which our information society is built, but the cable-cuts and other disruptions just do not stop. So despite best-efforts at physical protection, it seems to be one of those large-scale statistical certainties that a fairly high rate of cable cuts is inevitable.” With up to 100 Tbps of data flowing through a single fiber, “failures can have a catastrophic and far-reaching consequences. Physical failures of node infrastructures by fire or power loss is far less frequent, but software-related crashes and updates of routers within network nodes are a growing concern.”

1.2 Survivability in the Transport Layer

In the foreword to [Gro03b], Doverspike indicates that his “own work experience over the years demonstrates that efficiency in restoration and design of transport and packet client networks saves hundreds of millions dollars in capital expense in carrier networks,” and significant operational savings result from the increase in intelligence of network elements. So the information society evolves towards higher expectations of reliable telecommunication networks, where the transport network infrastructure will be almost invisible to the lay person because it is working nearly perfectly. But which of the service-level, the optical transport layer and/or the physical layer is better adapted for network protection purposes? Each layer in today’s network architectural model needs certain self-healing capabilities to recover from failures arising at their own or lower levels. When faults are successfully addressed at a given network level, upper layers are never aware of the lower layer outages. Fiber layer protection might prevent the propagation of cable damage, frequent at the physical level, by manually switching failed fiber equipment onto surviving fiber optics. But this manual process is typically too slow; so it is combined with higher layer automatic restoration. Which of the service or transport layers will make better combinations?

In the foreword to [Gro03b], Doverspike found it limiting to react to fiber optic cable cuts at the service level because the optical backbone transports its own switched voice and data overlay networks, as well as that of other network clients who have no knowledge of the physical layer although leasing the optical carrier to route their traffic. Conversely, he found it unreasonable to require that the service layer handles transport-level failures because OXC switching equipment is usually unknown and out of the control of the client carrier. Even if this was possible, our opinion is that it might be technically exhaustive to centralize the rerouting of the equivalent of 1.2 billions voice conversations affected by a single fiber failure. And following the argument of service-layer protection distributed among multiple carriers, the network will most probably incur congestion because paths will then be rerouted independently from each other. Considering increased transport network bandwidths instead, Doverspike's foreword to [Gro03b] indicates possible multiplex bundling and better economies of scale. Thus, the transport layer is of fundamental importance to invest with self-healing capabilities as it is clear that the automatic restoration of services affected by physical failures should be performed at the transport-level itself.

A widespread approach to prevent the event of failure in optical transport networks is to duplicate every transmission path, so that one is used under no failure conditions and the other in the restored network state. The literature proposes many survivability schemes, aimed at building such alternate disjoint routes for working demands to use under failure conditions. Typically, protection schemes are either in the form of rings or (dedicated/shared) mesh-routed, all of which create capacity redundancies across the network. Widely installed in the mid 1990s, ring-based topologies have simple and fast restoration properties, typically between 50 and 100 ms which is acceptable for most access technologies. But they impose specific routing patterns on normal state routes within the rings, for the purpose of loopback reaction to the event of failures, which results in quite capacity-expensive designs. Mesh-based survivability schemes, to which many networks migrate nowadays, allow more flexible routing of working paths. They operate with intelligent restoration principles that result in more capacity efficiency in the design. But intelligence in the routing of normal and restored state paths also brings more complexity in the design and slow restoration processes. Through this decade of ring versus mesh battles, it was unbelievable that, even theoretically, the same protection scheme could involve low working

over protection (or simply, “spare”) capacity ratio in the design and provide, at the same time, fast response to the event of failure preferably in 50 ms or less switching times.

1.3 The Case for p -Cycles

In the late 1990s, [GS98a] introduced a superior survivability technique named “ p -cycles” in reference to **p**re-configured, **p**re-connected **p**rotection cycles. The original intention was to build virtual rings in a full mesh context, thereby combining the real-time switching simplicity and speed of rings with mesh-like efficiency, flexibility and freedom in the routing of working and restored state paths. In the event of failure, p -cycles operate as a minor variation on rings; indeed, they provide the same straightforward failure location and fast switching, and they are not difficult to develop as an extension to the ring technology. Yet, in addition to the ring-like loopback reaction in the event of on-cycle failures, each protection channel on the p -cycle handles two working channels on every “*straddling span*.” This is of fundamental importance for optimization purposes because straddling spans themselves are not on the p -cycle, just their end-nodes are part of it. A straddling span bears no protection capacity, while having two protection routes on the two sides of the p -cycle in question. And unlike rings in which working paths are ring-constrained, p -cycles allow the use of entire mesh-like facilities in order to optimize normal state routing.

The discovery and understanding of p -cycles highlighted many other advantages. Conventionally, p -cycles follow the span-protecting paradigm with such advantages as locality of switching actions and span-like working envelope for demand uncertainty. But unlike other span-oriented architectures, span-protecting p -cycles have an intrinsic ability to also recover from node failures; in addition, the basic operating principle of p -cycles is extendable to the protection of path-segments and end-to-end paths. As well, p -cycles can be implemented either in a static or an adaptive way; and they can be configured at any level of the transport networking infrastructure (i.e. system, logical or service layer). On the other hand, the cycle geometric shape is much more manageable than linear greedy structures; so p -cycles support optimization-based methods that allow more capacity efficiency in the design. As a simple variations of ring-based architectures, p -cycles respect the legacy of ring-based equipment that was widely deployed before the advent of mesh-type

architectures; this ensures an easy and low-cost transition from SONET rings to mesh-type architectures.

1.4 Research Objectives and Highlights

An easy mistake would have been to rely on initial intuition and dismiss the change as having only minor potential effects and not pursue the fascinating topic of p -cycles further. After a dozen years of development, p -cycles are now well-established and competitive in the domain of network survivability. The literature is filled with material about p -cycles and an initial pursuit of their implications. However, the investigation of networking ideas, design methods and heuristics related to p -cycles is quite young relative to the protection schemes currently implemented in real-world carriers. As much network science on p -cycles is expected to follow, we now join the field, asking the following questions.

- i. Can a whole network be designed with p -cycles so that every path in the network is protected against both span and node failures? How much, if any, extra spare capacity would be required to achieve full node failure recovery in the design?
- ii. If all protected entities within a p -cycle network are segments of exactly two adjacent spans including their common node, is that alone enough to ensure the complete network protection?
- iii. How can we improve current design methods for failure-independent path-protecting (FIPP) p -cycles, and exploit this to node recovery purposes?
- iv. If it is ultimately glass that fails, what if just the glass is directly replaced? More specifically, what if p -cycles were used to rapidly, simply and efficiently provide for the direct replacement of failed fiber sections with whole replacement fibers?
- v. What is the implication of a capital expenditure (CapEx) cost exercise for p -cycles in the WDM layer? Does the assumption of capacity requirements in total channel-hops or channel-kms, typically used as cost metrics in network design, correlates real-world CapEx costs?

- vi. How can we elegantly control end-to-end path lengths in the restored network state? In doing so, what is preferable, minimizing the average length or the longest path, biasing the design towards choosing p -cycles that will minimize length (with or without no capacity penalty), or directly asserting a transparent reach limit?
- vii. Is there a way to solve a 200-node “large scale” network instance of any given p -cycle design problem, possibly involving many complicating practical constraints? What if a genetic algorithm (GA) evolves a preselected candidate population for a subsequent p -cycle ILP solution? What if the fitness function for the GA-like heuristic is the objective function of the p -cycle ILP itself!?

Thesis synopsis below summarizes studies conducted on the research questions above:

Chapter 2 introduces transport network geographical and layered architectural models.

It describes DWDM and OXC network elements, SONET/SDH/OTN interfaces to the optical layer, and the MPLS-based control plane. Chapter 2 also browses well-established and recent survivability schemes, with the aim of clarifying their difference with p -cycles. The end of the chapter focuses on the p -cycle survivability scheme: describes the operating principle and configuration in IP-over-WDM networks, discusses benefits and the conventional minimum spare capacity design model, provides a short literature review, recalls questions revisited and gives our research methodology.

Chapter 3 pertains to node-protecting p -cycle considerations. The need for node failure recovery is discussed, and prior related work is studied within a comprehensive literature review. Then, a general statement unifying all node-protecting p -cycle principles available within the literature is highlighted and exploited to derive a simple generalized approach to node failure recovery using ordinary p -cycles. The generalized operating principle is introduced and built into ILP mathematical design models for node failure recovery maximization, with or without no penalty over 100% span restorable minimum spare capacity designs, and for 100% node failure protection; resulting designs are compared to that of prior methods within a systematic study.

The two-hop segment protection paradigm is explored for the purpose of no longer distinguishing between node and span failures in p -cycle designs, but continuing to provide both capabilities. We explain how to transform a given network from the

two-hop segment protection perspective, how to map demand working routes on the two-hop segment graph and how to design two-hop protecting p -cycles for use under failure conditions. The overall concept is built into an ILP mathematical design model and a comparison is made with other p -cycle protection schemes.

The third study in this chapter pertains to the design of failure-independent path-protecting (FIPP) p -cycles. We provide some background on FIPP p -cycles, and investigate why FIPP design problems are especially quite complex to solve. General path-protecting p -cycles are introduced and exploited for an easier derivation of FIPP network solutions. The effectiveness of this novel approach is studied in regard to the results obtained using other FIPP-design methods in the literature, and the whole picture is applied to node failure recovery purposes.

Chapter 4 considers the question of restored state path lengths in p -cycle designs. The need for controlling optical reach and prior attempts to achieve this goal are first discussed. Then, we introduce a complementary matching of working paths with protection segments available through p -cycles, thereby controlling optical path lengths in the restored network state. This new matching principle is also built into several ILP mathematical models minimizing average and maximum path lengths into existing designs, biasing the design solution towards choosing p -cycles that will improve path lengths, or directly asserting a transparent reach limit. In closing the study, we address what is preferable between all of those ILP options.

As well, we discuss p -cycle configuration types in the DWDM optical layer, differentiating between opaque, hybrid and fully transparent networks. Recognizing the complexity of fully transparent p -cycle ILPs and identifying wavelength continuity requirements as being the cause, p -cycle protection is considered at the glass fiber level. Glass switched p -cycles and their motivations are explained and validated from the current-state-of-the-art fiber cross-connect technology. The concept of whole fiber switched p -cycles is built into an ILP mathematical design model and complexity improvements are evaluated vis-à-vis that of opaque, hybrid and fully transparent wavelength switched p -cycle ILPs.

The standardized NOBEL cost model for DWDM layers is also adapted to p -cycle specificities, taking into account the differences between opaque, hybrid and fully transparent network environments. A comparative study of real-world CapEx costs is then offered for wavelength and glass switched p -cycle architectures. In the experimental study, real-world CapEx cost evaluation differentiates between node, span and transmission pieces of equipment. As well, fiber utilization ratios and spare capacity requirement trends are compared to that of overall CapEx costs.

Chapter 5 pertains to the design of p -cycle networks on very large scales. We discuss the large scale p -cycle design problem and browse current practical methods that may lead to solve related ILP instances. We introduce a novel combination of genetic algorithms and integer linear programming, referred to as GA-ILP, which has many features to recommend it for solving any large scale p -cycle design problem requiring the pre-selection of a very few candidates from an almost infinite space. In further considerations, the GA-ILP is used to provide the world's largest solved instance of p -cycle design problem, at a 200-node sheer scale. And the GA-ILP is repeatedly required to solve large instances of more complex p -cycle models developed under advanced design considerations within the thesis.

Chapter 6 summarizes each of the thesis' studies and the main research findings. Then, innovative aspects of those contributions are highlighted and their implications are discussed. The chapter ends with an overview of other studies conducted during our PhD program, possible lines of future research directions, and the outline of publications associated with the thesis' series of studies.

Bibliography and Appendices close the overall thesis. Appendix A portrays a conventional p -cycle minimum spare capacity design solution and proves its validity. Appendix B provides ILP mathematical design models for opaque, hybrid and fully transparent wavelength switched p -cycles.

Chapter 2

Background on IP-over-WDM and p -Cycle Survivable Networks

¹ This chapter aims to put the research performed for this thesis into context. Section 2.1 presents the concept of transport networking and discusses the specific case of “*IP-over-WDM*” networks. We touch on layering and partitioning aspects, describe network elements and the technology involved, and give the terminology typically used in this field area. Section 2.2 justifies the need for survivable transport networks, explains the measures of survivability, and browses numerous survivability schemes. Section 2.3 focuses on the p -cycle protection technique: it provides a general understanding of p -cycle benefits and protection types, discusses possible configurations in IP-over-WDM networks, and gives some background on the design of p -cycle survivable networks. Section 2.4 gives a short literature review, recalls questions addressed in this dissertation, and presents our research methodology.

2.1 Introduction to Transport Networking

Telecommunication networks can be partitioned into multiple geographical tiers as shown in Figure 2.1. At the edge of the hierarchy, the *access* tier connects end-users to telecommunication central offices (COs). Access tier networks collect and distribute traffic of all sorts (i.e. data, voice and video) from/to the end-users. Each access network handles tens to hundreds of end-users, such as residential and business customers or cell companies’ base stations, for a few kms. The metropolitan or simply “*metro*” core tier interconnects COs at

¹The entire chapter is adapted from [Gro03b], which is a thorough reference book on mesh-based survivable transport networks, and from the ECE-681/EE-780 UofA class notes provided by [Gro02a, Dou06].

2. Background on IP-over-WDM and p -Cycle Survivable Networks

distances of tens to hundreds of kms. Any given metro core network comprises a number of COs, typically owned by the same carrier, a few of which are connected to both access and long-haul tiers. The aim is to aggregate the traffic coming from CO-related access networks, i.e. thousands of end-users, and moves this either internally between COs or out of the metro area in question through the long-haul tier. Long-haul networks interconnect metro networks, smaller community COs, Internet service providers (ISPs), and regional and international gateways for millions of end-users and typically over thousands of kms.

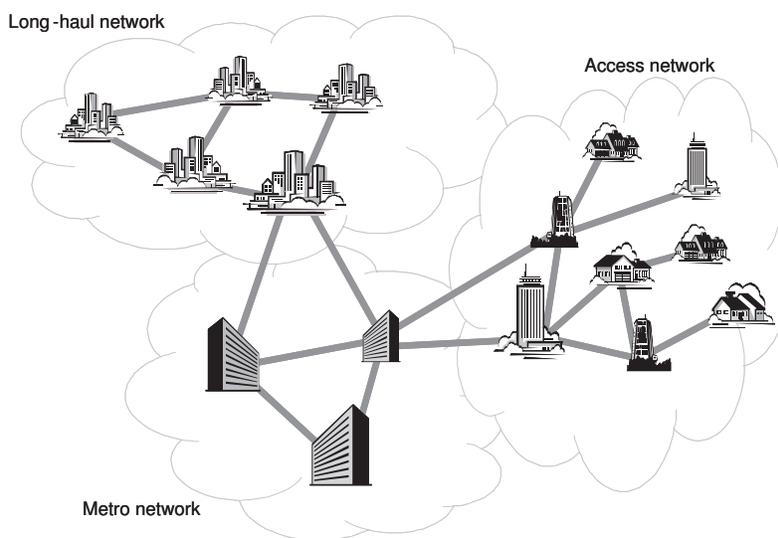


Figure 2.1: *Segmentation of Telecommunication Networks in Multiple Geographical Tiers, adapted from [Gro02a, Gro03b, Dou06].*

In the geographical network hierarchy above, the end-user naturally perceives and interacts with access technologies, such as the Internet and circuit-switched telephony, as if they were separate physical networks at all tier levels. But there is no cable between IP routers or voice circuit switches. Rather, Figure 2.2 shows how all specialized services are multiplexed to form a set of standard-rate digital carrier signals operating over a common backbone, either in the metro core or the long-haul tiers. So all of the user-perceived networks are just logical abstractions created within one physical infrastructure, the *transport network*. This transport network consists of an essentially fixed set of multi-channel point-to-point transmission systems, borne on *fiber optic* strands and which are managed to create virtual network environments for all other services [Gro03b].

2. Background on IP-over-WDM and p -Cycle Survivable Networks

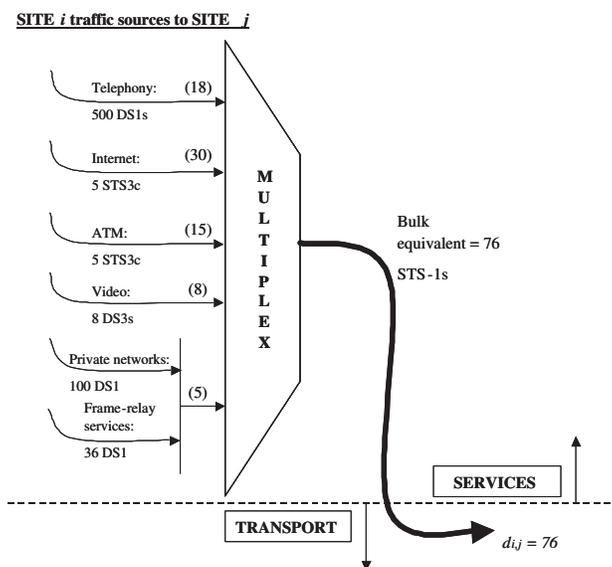


Figure 2.2: Aggregation of Service Layer Traffic into Transport Demands, adapted from [Gro02a, Gro03b].

2.1.1 Layered Architectural Model

Figure 2.3 gives another useful stratification of service over transport networks. This is a three-layered architectural model comprising a service layer at the top of the hierarchy, a fixed geographical facility route layer at the bottom, and a transport level in between them with two sub-layers: the transmission system layer and the logical cross-connect layer. In another partitioning view, the service and the logical layers sit above the physical layer which comprises the geographical facility route and the system layers.

Each of the layers in both hierarchies has a native form of demand units that it aggregates into the form of capacity units for the next lower layer. So any layer has its own generic *node* and *link* resources: nodes provide points of access to the next upper level, and perform switching and routing operations. A link or *channel* provides a single unit of bandwidth to carry demand units, at the respective level of transport management. A concatenation of cross-connected links forming a unit-capacity digital connection between two end-points (i.e. end-nodes) is referred to as *path*; and a set of link-disjoint paths sharing the same end-nodes is called *pathset*. More specifically, the following provides a snapshot of basic elements and related capabilities at each network level.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

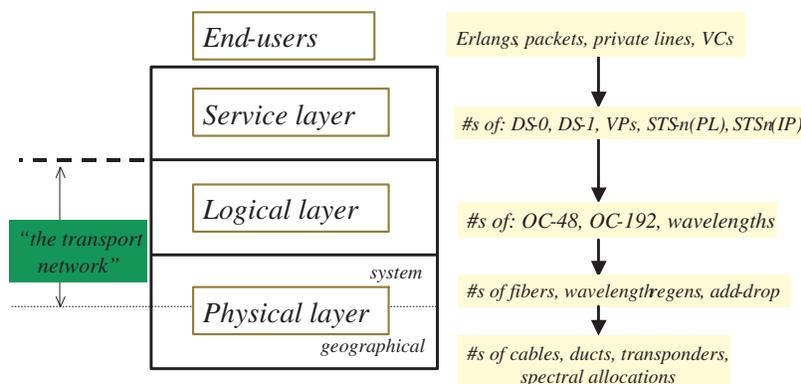


Figure 2.3: Layered Architectural Model, adapted from [Gro02a, Gro03b].

The service layer consists of a collection of user-perceived networks. The logical model for all of those networks still holds as if each of them has its own dedicated transmission system. But Internet connections, individual telephone calls, packet streams, leased lines, and so on, cannot make their own way natively over fiber transmission systems. Rather, IP routers, label switched routers (LSRs), telephone switches and other service-level nodes aggregate Erlangs, packets, private lines, etc., to form a set of standard-rate digital carrier signals of such rates as DS1, DS3, STS1, OC-3, OC-12 and GbE.

The logical cross-connect layer is responsible for services *grooming*, logical transport configuration, and bandwidth allocation and management. Grooming means aggregation of lower rate service payloads into higher-rate outgoing signals to allow more efficient use of high speed facilities. It is optical and digital cross-connects (i.e. OXCs, DCS) that aggregate capacity units such as DS1, DS3, STS1, OC-3, OC-12 and GbE above, in order to fill standard-rate containers created in the transport backbone. Demand units generated are typically OC-48, OC-192, wavelengths and wavebands.

The transmission system layer provides point-to-point bit-transmissions at 2.5 to 10 Gbps, point-to-point fibers, or point-to-point wavelengths between terminal multiplexers. Demand units generated are typically at the fiber or cable levels of granularity.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

The **passive physical sub-level**, referred to as **geographical facility routes** layer, provides the medium of transmission connectivity. It essentially provides right-of-way, conduits, pole-lines, cables, ducts, etc., between buildings, man holes or equipment huts. In transport networking terminology, a *span* denotes a set of links between two nodes that are adjacent in the physical graph; and a *route* is a set of span designations that are contiguous on the physical graph.

2.1.2 Towards IP-over-WDM Networks

With a yearly growth rate of 100 to 200%, traffic volumes from the Internet and the revolutionary applications it enables finally surpassed all other forms of traffic. Remaining services which now make a smaller fraction of the total traffic, typically 3 to 5%, are also converted to IP packets for switching and transport which give rise to the IP-over-optics paradigm [Gro03b]. Despite the implications of the simplified phrase “IP-over-optics” that IP packets might be directly applied to a laser transmitter, there are actually quite stringent requirements for an arbitrary bit sequence to be reliably transmitted and received over a physical layer path. There is always a need for some type of synchronous bit or byte timing for regeneration, certain transmission coding properties for receiver decision threshold control, an assured bit transition density for low-jitter clock recovery, and so on.

Many stacks of multiplexing and transmission technologies correspond to the view of IP over transport networks. But the conventional layering adopted in the late 1990s is a stacking of: (iv) IP for user applications and local area network (LAN) environments; (iii) asynchronous transfer mode (ATM) for virtual circuit/path capacity engineering, flow control, performance monitoring, virtual networking and quality-of-service (QoS) guarantees in the data networking layer; (ii) synchronous networks (SONET) for high quality transport of payloads over the fiber physical medium, error monitoring, management, synchronization and protection switching; and (i) dense wavelength division multiplexing (DWDM) for sheer capacity and an effective multiplication of fibers in the ground. However, this four-level model poses many problems such as the complexity of interlayer interfaces, configuration details, management aspects of each individual layer, as well as the total amount of space, power and equipment spares and repair costs involved to sustain all the layers, and an inevitable loss of capacity efficiency from so many layers.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

As revenues from data are actually much lower than those from voice and other service connections, network operators are looking for ways to support the IP data growth with less equipment investment, more flexibility, and more provisioning speed. Since 2003, IP-over-optics have tended to migrate towards a 2.5 layered-architectural model: generalized multi-protocol label switching (GMPLS) is integrated with the IP service layer to assume the role of ATM and controls circuit establishment at the SONET and DWDM levels. Most of the SONET duties are referred down to the DWDM transmission system layer; and either a thin but enhanced SONET or the optical transport network (OTN) interface handle such functionalities as forming bit streams for physical transmission, framing and error monitoring which cannot be eliminated. The following gives a general understanding of SONET, OTN, DWDM and GMPLS network technologies.

2.1.3 Synchronous Optical Network (SONET) Standard

Standards for synchronous optical network (SONET) and its European counterpart synchronous digital hierarchy (SDH) were primarily developed for interoperability purposes between separately purchased piece-parts of transmission systems such as transmitters, receivers, regenerators, cross-connects and network controllers, all of them from different vendors. In addition to the standardization of transmission systems using signal formats, monitoring methods, modulation techniques, laser types, and so on, that were initially proprietary to each vendor, SONET/SDH plays an important historical and ongoing role in optical transport networking. SONET actually has an ability to keep payloads as much as possible in the native format and structure in which they exist in the LAN environment, which is the idea behind multiple client service networks over one single backbone. In the specific case of IP-over-WDM networks, the SONET technology is enhanced with the generic framing protocol (GFP) which is a standard way to adapt any frame-oriented packet data sequence or byte-oriented data signal for transport over a suitably sized SONET envelope. These attributes help to eliminate the latency, power and space consumption, and complexity of management, of going through a stack of adaptation protocols.

SONET networks provide all of the basic logical elements for monitoring, fault section-alization, voice orderwire, protection switching, remote provisioning, etc. The basic frame structure is the synchronous transport signal-level 1 (STS-1), with a *timeslot* of 125 μ s,

2. Background on IP-over-WDM and p -Cycle Survivable Networks

reflecting the basic period for sampling speech at 8,000 times a second. STS-1 specifically provides transport overhead and a synchronous payload envelope (SPE) in nine rows by 90 columns of 8-bit bytes. The transport overhead occupies the first three columns of the frame and is further divided into line and section overheads that provide signal framing, line identification, performance monitoring, and voice and data channels used for provisioning and maintenance. And the SPE occupies the remaining 87 columns by nine rows; the first column is used for path overhead functions such as end-to-end performance monitoring and path identification while the other 86 columns are for payload signals.

With a total of 810 bytes, STS-1 has a bit rate of 51.840 Mbps. SONET signals of higher rates, i.e. STS-3, STS-12, STS-24, STS-48, STS-192, etc., are all multiples of the STS-1 rate. They are obtained by simple assembly of a whole number of intact and separate STS-1s, each with its own payload and overheads. Services that require a clear-channel multiples of the STS-1 payload (e.g. STS-3c for ATM) can also be transported by concatenating several STS-1 signals together, but with the difference that the resultant signal STS-Nc has a single N times payload field and a single overhead stream. In both cases, an STS-N signal is converted to the corresponding optical carrier signal (OC-N) prior to transmission. Contrary to the *increasing* rate for SONET/SDH digital signal hierarchy, some applications rather require a finer granularity of transport bandwidth. Another enhancement to SONET is the *virtual concatenation* (VC) technique that provides a much wider family of SPE rates, moving the concatenation capability down to the *virtual tributary* (VT) level. The respective transport capacities, referred to as “VT1.5- nv ” or “STS-1 nv ,” make bandwidth fairly efficient for such payloads or applications as Ethernet at 10 Mbps (VT1.5-7v), fast Ethernet at 100 Mbps (VT1.5-64v) and GbE at 1000 Mbps (STS-3c-7v or STS-1-21v).

The generic node for SONET networks is the digital cross-connect system (DCS), which can be thought of as a circuit switch for digital carrier signals. A SONET DCS accepts various electrical and optical carrier signals, accesses the individual tributary signals (e.g. VTs, STS-1s, STS-Nc) in electrical form and switches them from incoming to outgoing facilities. Note that none of the payloads of those transport signals is accessed, manipulated or altered; only the routing of the signal is affected. In addition to this switching function, a SONET DCS provides local add/drop and multiplexing/demultiplexing functions. The market distinguishes between the pure add/drop multiplexer (ADM) that is a DCS of two

2. Background on IP-over-WDM and p -Cycle Survivable Networks

line-rate interfaces typically used at intermediate sites along linear add/drop chains, the basic *broadband* DCS (B-DCS), and the *wideband* DCS (W-DCS) mainly used for grooming purposes. But it was as earlier stated that nowadays, DWDM, OTN and GMPLS assume many or all of the SONET functionalities in the IP-over-WDM networking paradigm.

2.1.4 Dense Wavelength Division Multiplexing (DWDM)

Dense wavelength division multiplexing (DWDM) technology provides a way to multiply the physical layer fiber infrastructure to cope with the growth of traffic. The technique is to divide any given fiber optic based on tightly spaced and controlled optical carriers in the 1500 nm range, where optical fiber attenuation is at its lowest. This enables the simultaneous transmission of many client digital signals onto a single fiber optic by assigning to each of them a unique optical carrier, referred to as *wavelength*. The number and selection of wavelength channels implemented depends on the application requirements such as optical reach, data rate on the carrier, fiber type, optical filter technologies used, etc. But typically, available DWDM systems support up to 64 wavelengths per fiber optic at 2.5 Gbps in metro area networks, versus 160 wavelengths in long-haul and ultra-long-haul (ULH) networks at 10 Gbps and 2.5 Gbps respectively. Systems operating at 40 Gbps are emerging and higher bit-rates of 80 to 160 Gbps are under research and development.

2.1.4.1 Basic Network Elements

In DWDM, any wavelength channel is equipped with a light source, typically a *laser*, and with a photodetector. The laser converts any input signal in the electrical form into an optical signal so that it can be transmitted over the fiber medium; conversely a *photodetector* converts any optical signal into an electrical signal if needed. In between the laser at the source and the photodetector at the destination, the optical signal accumulates such impairments as attenuation loss, dispersion and nonlinear effects. An *optical amplifier* is deployed at DWDM node sites and every 80 to 120 fiber-km to counteract attenuation losses over a complete band of lightwave (i.e. wavelength) channels at once, without demodulating or in any way individually processing each optical carrier. To eliminate other optical component related impairments, any in-line amplifier is equipped with a dispersion-compensating fiber; and the complete waveband goes through a dynamic gain equalizer every fourth in-

2. Background on IP-over-WDM and p -Cycle Survivable Networks

line amplifier. But amplifiers add noise, and even boost any incoming noise along with the signal: *regeneration* is required to counteract the effects incurred. In the regeneration process, each channel is individually demodulated from its optical carrier, converted in its pulse form electrically for retiming and reshaping, and remodulated for transmission onto a new outgoing carrier wavelength.

2.1.4.2 Optical Cross-Connect (OXC)

Regeneration is a per-channel process involving optical-electronic-optical (o-e-o) conversion and high speed electrical processing of each channel individually and is therefore more costly than optical amplification. Often the regeneration function needed along an optical path would be provided by an *optical cross-connect* (OXC), which is the counterpart of DCS and the generic node for DWDM layers. The OXC technology typically involves a micro-electro-mechanical system (MEMS) which is a miniature movable mirror of 1000 input ports and 1000 output ports, with an ability to deflect in about 10 ms a light beam from each input port to the desired output port. From the operational perspective, Figure 2.4 pictures the OXC as a circuit-switch that connects from any input fiber to any output fiber and performs switching from one wavelength to another, as well as from one fiber optic to another in space. Any DWDM client signal adds and drops the OXC via an optical *transponder* that electrically regenerates the signal and translates it to a specific wavelength channel. So transponder devices can also serve for wavelength conversion purposes.

On the basis of transponder availability, one may distinguish between three OXC types: OXCs providing full regeneration and wavelength conversion capabilities through transponder devices placed on every outgoing port, OXCs offering partial wavelength conversion through a limited pool of transponders, and pure OXCs with no wavelength conversion or regeneration capability having only add/drop transponders. Those three basic types of OXCs give rise to concepts of *transparency*, *opacity* and *translucency*. In transparent optical networks, every node is a pure OXC with no regeneration or wavelength conversion capability; each lightpath is thus carried from its source to the destination without any electronic processing at intermediate nodes en route. The opposite of transparency pertains to the concept of an opaque optical network, where all OXC nodes are capable of full regeneration and wavelength conversion; any incoming lightpath can be transformed

2. Background on IP-over-WDM and p -Cycle Survivable Networks

to any other wavelength on any outgoing fiber through o-e-o conversion and remodulation on a new laser. Translucency strikes a balance between transparency and opaqueness, by dividing a large-scale optical network into several “transparent islands” and confining regeneration and wavelength conversion functionalities at transparent domain boundaries.

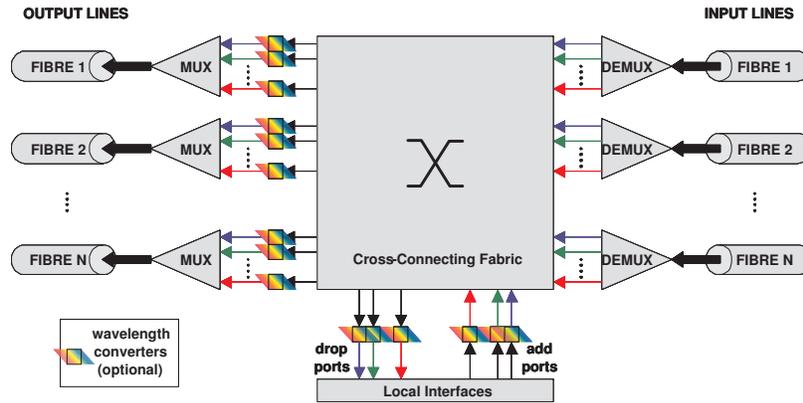


Figure 2.4: Functional Block Diagram of an OXC, adapted from [Dou06, Gro03b].

2.1.4.3 Transparent Optical Networks

The principal characteristic of a transparent optical network is that every node within the network is a pure OXC, with no regeneration or wavelength conversion capability. Each optical signal is carried through a *lightpath* that connect its source to its destination without any electronic processing at intermediate nodes en route. All-optical applications and the proliferation of fiber optic technology into various markets offer many benefits. The use of fiber optics to relay data signals over long distance is now widespread. All-optical switching eliminates as well the need for o-e-o conversion, offering the ability to switch optical signals transparently and independently of data rates, formats, wavelengths, protocols and services. And all-optical switching provides network operators with much needed automation capabilities to create, monitor and protect optical paths.

Despite a possible cost-effectiveness due to the absent o-e-o, limitations and issues in network management make fully transparent networks difficult to implement and manage. Such an issue relates to the unavailability of all-optical wavelength conversion in current

2. Background on IP-over-WDM and p -Cycle Survivable Networks

OXCs. Subsequently, achieving the goal of transparency requires a wavelength assignment that must be uniquely reserved for each given path on any fiber ridden en route. In addition, it is not possible to indefinitely preserve all attributes—e.g. phase, frequency, waveshape, amplitude, polarization, and most important of all, signal-to-noise-ratio (SNR)—of a signal transmitted over an increasing length of fiber and number of optical amplifiers and possible wavelength-changing transponders. Thus, the distance limit is another fundamental to retain required SNR, because a fully transparent network does not provide electronic regeneration en route. Typical reach limits are of 75 km for metro area network, 400 to 600 km for long-haul networks, and up to 2500 to 4000 km for ULH.

2.1.5 Generalized Multi-Protocol Label Switching (GMPLS)

IP routers, SONET DCS, OXCs, DWDM systems, etc., all use generalized multi-protocol label switching (GMPLS) to dynamically provision timeslots, wavelengths and fiber resources, and to accommodate network survivability techniques. The following gives an overview of the original multi-protocol label switching (MPLS) and enhancements in an IP-integrated layer. Then, the discussion extends MPLS attributes to encompass time-division, wavelength and spatial switching from one fiber to another and addresses open possibilities for a GMPLS control plane of the entire transport networking infrastructure.

2.1.5.1 IP/MPLS Service Layer

In IP-over-MPLS networks, ingress *label switched routers* (LERs) look at the destination IP-address in the header of any packet first arriving into the MPLS domain. The ingress LER selects an appropriate label for the packet based not just on that destination information but also on quality-of-service (QoS) considerations and even explicit routing requirements. Labels will be removed by the edge LER through which the packet will leave the MPLS domain. Once the packet is encapsulated with a label and put in the appropriate outgoing queue, it is like the packet has been dropped into a logical pipe or tunnel directed to its destination. All subsequent *label switching routers* (LSRs) in the MPLS core forward the packet based on the incoming port and packet's label informations. But prior to the packet forwarding, the LSR that currently processes the packet changes the value of the label to indicate the node to which the next LSR should pass the packet.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

The sequence of relays represented by “ $\{in\text{-}port, label\}$ to $\{out\text{-}port, new\ label\}$ ” at every LSR between the ingress and egress nodes defines a *label switched path* (LSP). However, packet traffic does not constitute a significant flow per se to warrant an LSP establishment. To encounter this, LERs use the concept of forwarding equivalence class (FEC), which consists of bundling together IP destinations that share the same local routing decision from the standpoint of the given entry node, and keep a label information base (LIB) recording of what initial label and outgoing port define the correct LSP to assign to each bundle. The establishment of coherent LSPs also requires a constant LIB at all LSRs within the network. Label switching relationships can be either written in pre-computed and centrally downloaded label swapping tables; or they can be requested and released on demand using label distribution protocol (LDP).

The manipulation of logical *circuit-like* quantities through MPLS is fundamentally much faster, autonomous and reliable than the redirection of single packets through conventional IP routing tables. MPLS also constitutes a more scalable solution than IP datagram routing, even with label swapping tables, as MPLS operates data forwarding between neighbour-nodes only—which does not require a correct routing table entry for every possible destination address in the administrative area, as in pure IP. In addition to the initial need for a circuit-like logical construct for such networking purposes as delay throughput, loss rate and scalability improvements, MPLS allows in practice quality-of-service (QoS) assurances, load balancing over two routes, and effective and fast schemes for survivability.

2.1.5.2 Generalized MPLS and Enhancements

ATM also uses the concept of label switching with possibly the same benefits as MPLS. The relevance of MPLS to IP-over-WDM transport networks is not in packet forwarding itself, but rather the whole philosophy for LSP establishment can be fairly naturally generalized to the layer stack through GMPLS. When we begin to consider SONET timeslots or DWDM circuit establishment under GMPLS, the label sequence becomes a specification of literally what timeslots, wavelength channels or fibers we want cross-connected to realize a hard physical circuit, rather than strictly just a logical LSP. Label swapping table equivalences are simply the output and next timeslot or wavelength stored in the physical connection state, between input and output ports in a DCS or an OXC. Once those

2. Background on IP-over-WDM and p -Cycle Survivable Networks

switching relationships are set up, the result is an inescapable sequence of relaying actions that direct any input to the pre-determined output in a completely circuit-like way.

GMPLS is much more than just an interesting analogy between LSPs and lightpaths or timeslots. The practical advantage is that by identifying and treating all transmission resources as labels, GMPLS allows all forms of path setup to follow fairly simple extensions of the same logical process and protocol implementations and to use the same form of databases as MPLS. Combined with the MPLS-like label stacking feature, GMPLS opens the way for many interesting possibilities such as a type-consistent and top-to-down hierarchical label distribution at each node; the aggregation of small granularities such as a set of LSPs or lightpaths sharing different end-nodes, into a high capacity LSP or single waveband path over a common segment; the abstraction of entire subnetworks to single hops, thereby allowing routing through various regions or network domains where the ingress and egress points are known but the internal routing details of the domains are not.

2.1.6 The Next Step in the Optical Network Evolution: OTN²

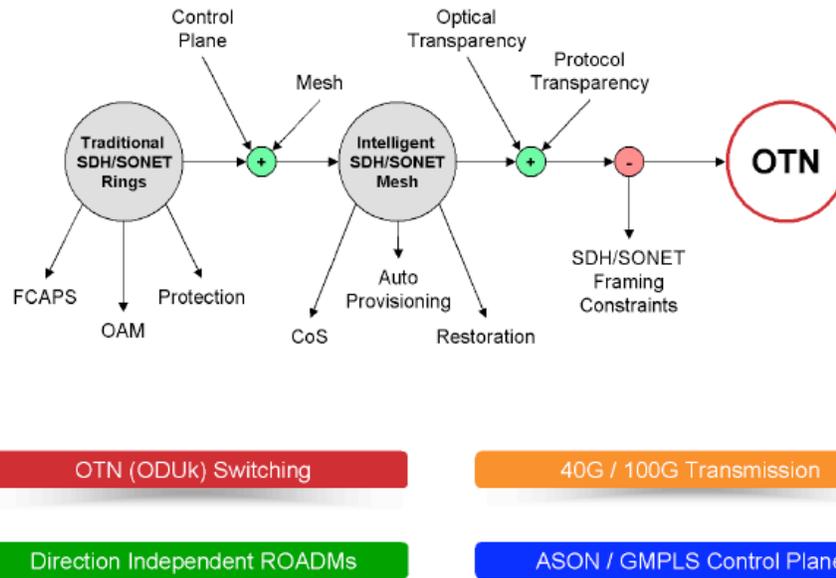
Instead of using a thin and enhanced SONET/SDH interface for the mapping of IP/MPLS service-level payloads onto the optical transport layer, DWDM networks are now standardized by the functional architecture called optical transport network (OTN). To state it in a summary form, the OTN technology is defined by the ITU-T's G.709 recommendation and belong to a protocol for the transmission of a multitude of services over fiber optics with a high degree of flexibility, resiliency and manageability. It provides a means to encapsulate and transport any traffic type over optical wavelengths, thereby offering a graceful migration path from SONET to Ethernet while maintaining the advantages of both. It also allows great deployment flexibility that would save money, mitigate risk and expedite revenue to any network operator looking to grow his backbone capacity and capabilities.

2.1.6.1 Why OTN, in place of SONET/SDH?

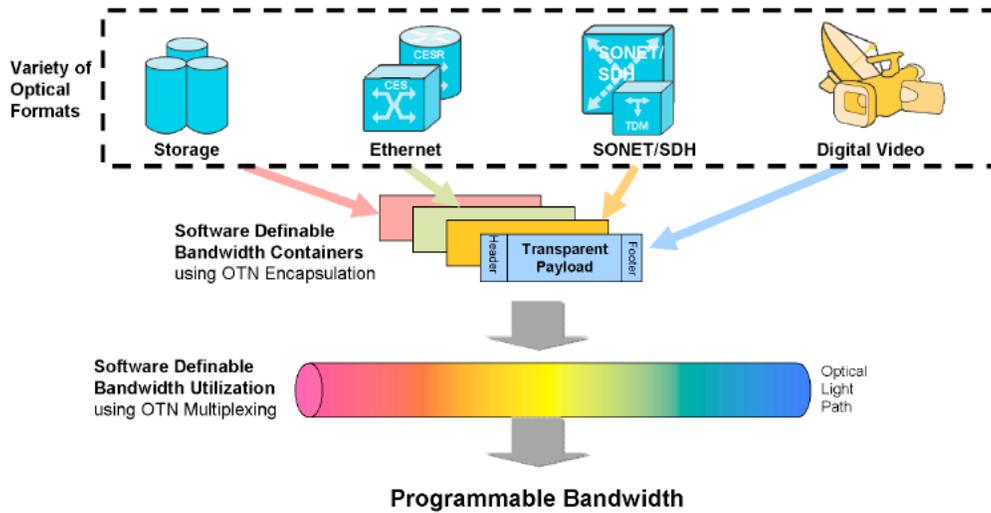
[BJ10] gives more details on the topic of OTN. He outlines the basic construction of an OTN-compliant transport network and discusses its capabilities and benefits to network

²This section is entirely adapted from the Ciena virtual seminar about “*The Next Step in the Optical Network Evolution: OTN*,” given on 9th February 2011 by Loudon Blair (Sr. Technology Director at Ciena Corporation) and Chris Janson (Sr. Product Manager at Ciena Communications)—[BJ10].

2. Background on IP-over-WDM and *p*-Cycle Survivable Networks



(a) Next Generation of Optical Transport



(b) Bit Transparent Transport of Client Signals

Figure 2.5: OTN: an Evolution Platform for Multiple Services, [BJ10].

2. Background on IP-over-WDM and p -Cycle Survivable Networks

operators. One substantial advantage is that OTN provides a robust multi-service transport infrastructure, as shown in Figure 2.5(a). Active standards activity at ITU demonstrates widespread adoption as a framework for WDM networking, defined with close cooperation with IEEE 802.3 standardization for 10/40/100 GbE. Even though those bit rates are synonymous with Ethernet, Figure 2.5(b) shows that the OTN protocol is an efficient way to map many different service types on to a single wavelength.

On the other hand, OTN is lower cost and much less complicated than SONET/SDH:

- The OTN technology uses an asynchronous mapping of payloads where SONET/SDH requires a synchronous transfer mode.
- Aimed at providing a functional architecture for DWDM networks, OTN is designed to operate on multiple wavelengths and to scale to 100 Gbps (and beyond) in support of 100GbE. This stands in contrast to SONET/SDH that was originally specified to operate on a single wavelength of no more than 40 Gbps.
- OTN processes high traffic volumes with a single-stage multiplexing versus multi-stage multiplexing performed with SONET/SDH.
- With OTN, a standard forward error correction (FEC) solution is necessary to transmit rates of 10 Gbps or more over any reasonable distance. This is a Reed-Solomon FEC RS (255/239) that uses a fixed frame size, sized for error correction of 16 blocks per frame for correcting 8 bit errors per block, and increases the frame rate as the speed increases. Conversely, SONET/SDH uses a fixed frame rate and increases its frame size with the speed; this corresponds to one frame per 125 μ s (or 64 kbps voice data rate for 8000 bps.)

2.1.6.2 How It Works?

OTN is truly understandable through its similarities and improvements to SONET/SDH. Next, we consider the four data processing steps in Figure 2.6—i.e. framing, encapsulation, error correction and monitoring.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

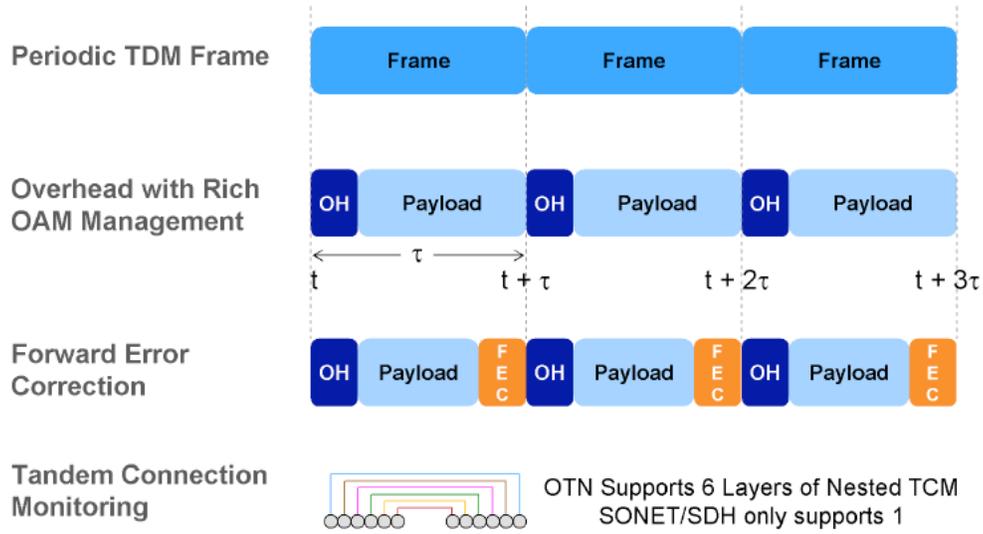
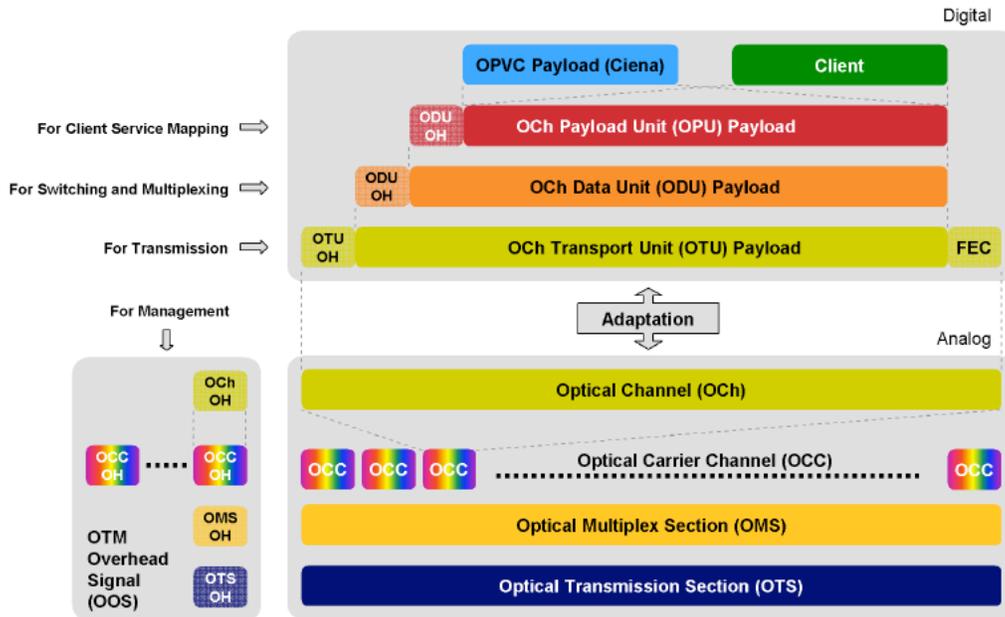


Figure 2.6: *Similarities and Improvements to SONET/SDH, [BJ10].*

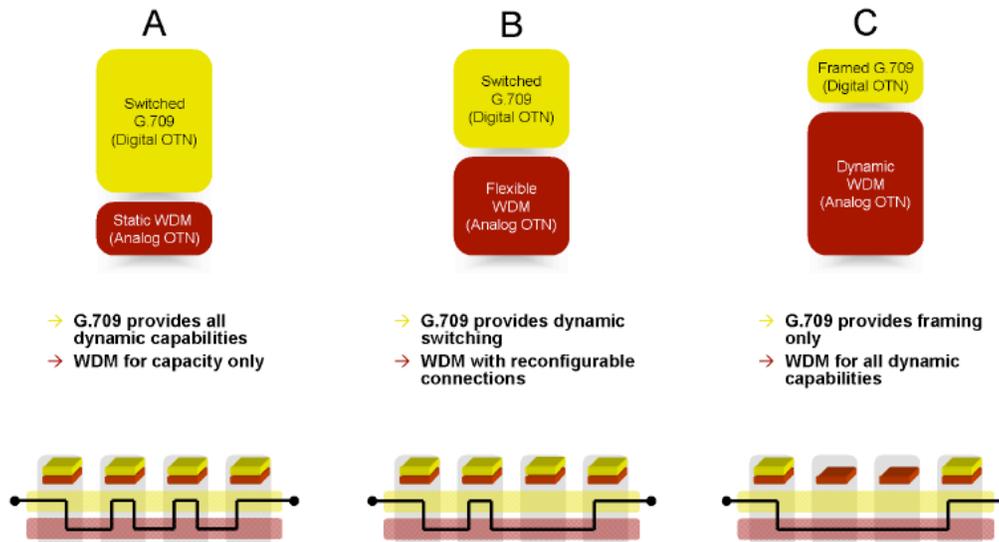
Transport Module and Multiplexing Figure 2.7(a) gives the optical transport module (OTM) for OTN. Accordingly, OTN technology supports up to six sub-layers of nested tandem connection monitoring (TCM). The optical/analog domain is dedicated for management purposes and comprises an optical transmission section (OTS) that consists of unverified physical links, an optical multiplex section (OMS) for wavelength selective bypass, and an optical channel (OCh) for full wavelength switching. The electrical/digital domain contains the three other layers: i.e. OCh transport unit (OTU) aimed at pre-establishing physical links, OCh data unit (ODU) capable to offer high grooming efficiency, and OCh payload unit (OPU) for client service mapping and sub-wavelength ODU switching. Optical and electrical domains work together, using any architecture option in Figure 2.7(b).

OTN Management Supervision OTN overhead is rich of OAM management functionalities. Figure 2.8(a) illustrates a powerful network monitoring, which belongs to frame alignment, payload management, path monitoring, TCM and section monitoring. Key capabilities include: “*continuity supervision*” that monitors the integrity of a link; “*connectivity supervision*” that monitors the integrity of a sequence of connections by com-

2. Background on IP-over-WDM and *p*-Cycle Survivable Networks



(a) Optical Transport Module



(b) Three Architecture Options

Figure 2.7: Optical Transport Module, [BJ10].

2. Background on IP-over-WDM and p -Cycle Survivable Networks

paring source and destination IDs; “*signal quality supervision*” that monitors performance error parity check on frame before and after transmission; “*payload mismatch supervision*” that monitors for correct client payload at source and destination by matching payload type; “*alignment supervision*” that monitors alignment of OTN frames; “*maintenance information*” that suppresses alarm escalation by informing upstream/downstream of defects, provides qualification information for alarm, supports single-ended supervision of a connection; and “*management communications*” that provides communications channels for path, section and out of band management communications.

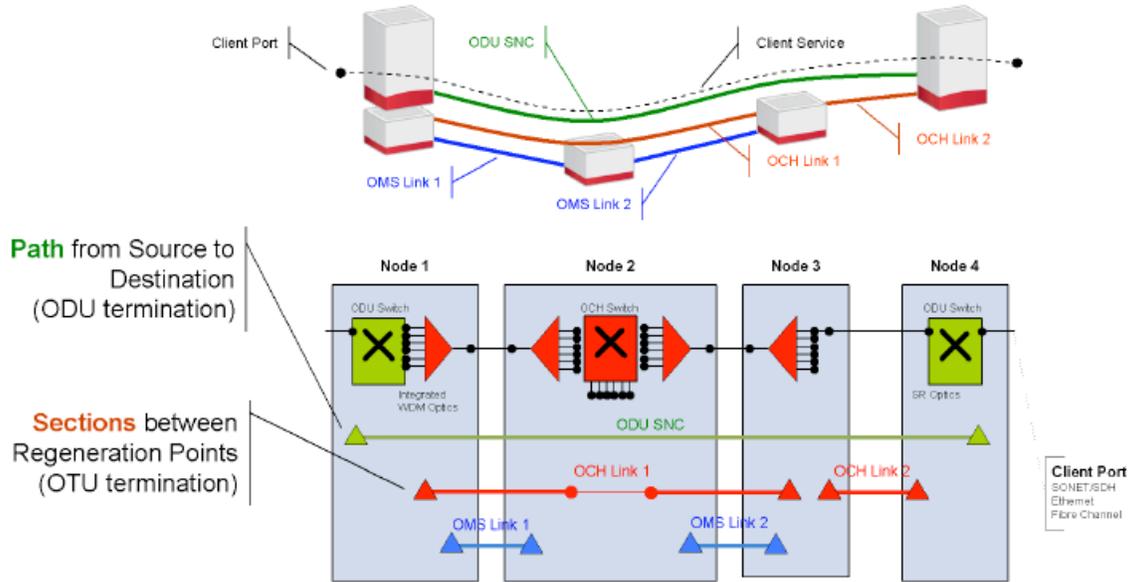
Figure 2.8(b) also shows that TCM provides management visibility at multiple (nested) levels. This allows to perform a single-stage multiplexing that was not possible with SONET/SDH, originally conceived to support only one TCM layer. For reference, Figure 2.9(a) illustrates the OTN one stage multiplexing structure used before 2009 and compares this with SONET multi-stage multiplexing. And Figure 2.9(b) shows post-2009 OTN multiplexing structure.

Frame Structure and Bitrates Figure 2.10(a) provides full OTN frame structure. Given a payload for client signal, OTN adds 1 overhead byte for every 238 payload bytes, and 16 FEC bytes for every $\{238 + 1 = 239\}$ bytes. Resulting 255 bytes format is repeated 16 times per row for 4 rows, corresponding to a total of 16320 bytes per frame. Regarding bitrates, Figure 2.10(b) indicates G.709 defines rates and non-standard supplemental rates based on (255, 239) FEC coding.

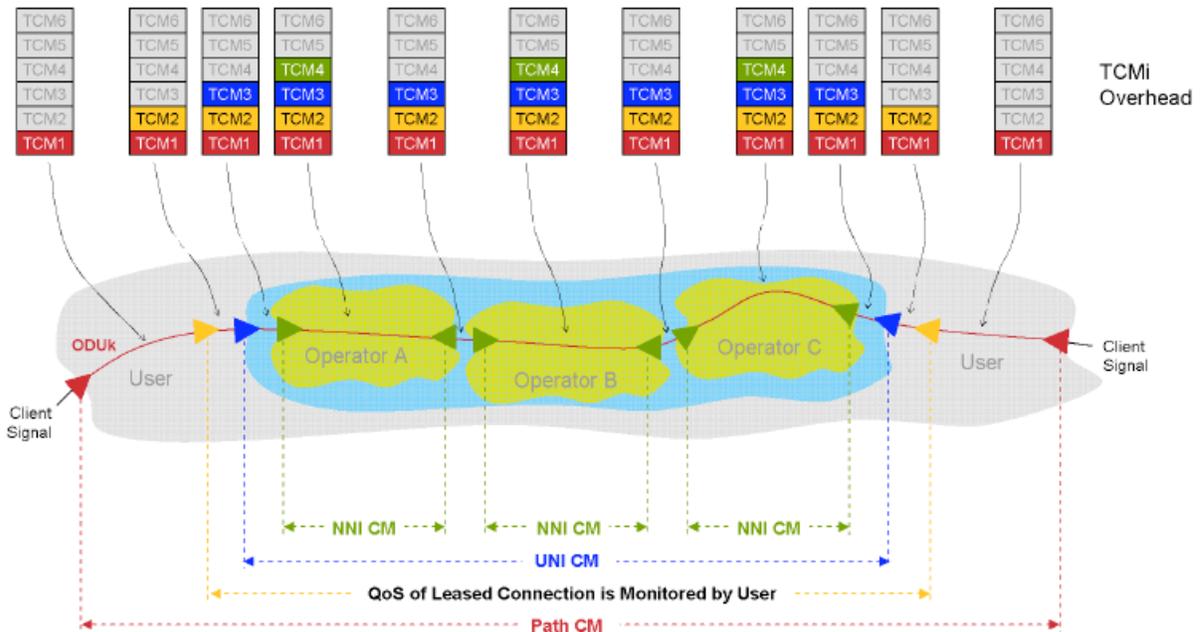
Forward Error Correction Roughly 6% of each OTN frame is dedicated to an error correcting code. This is to add redundancy to the message in question through encoding prior to transmission to enable the receiver (decoder) to correct errors induced in the communication channel. It results in a roughly 6dB coding gain for an OTN signal with choice of lower error rates, lower transmission power and greater transport distance.

In practice, the OTN standardization uses the Reed-Solomon (RS) code 255/239 given in Figure 2.11, with 239 base data bits and 16 added overhead bits (i.e. 6.7% overhead). This allows the correction of 8 or less bit errors in 239 bits, which corresponds to $8 \times 16 \times 4 = 512$ bits per OTU frame. Anything over 8 bits is completely uncorrected, and the typical gain is ~ 6.5 dB at 10^{-12} BER [BJ10].

2. Background on IP-over-WDM and *p*-Cycle Survivable Networks



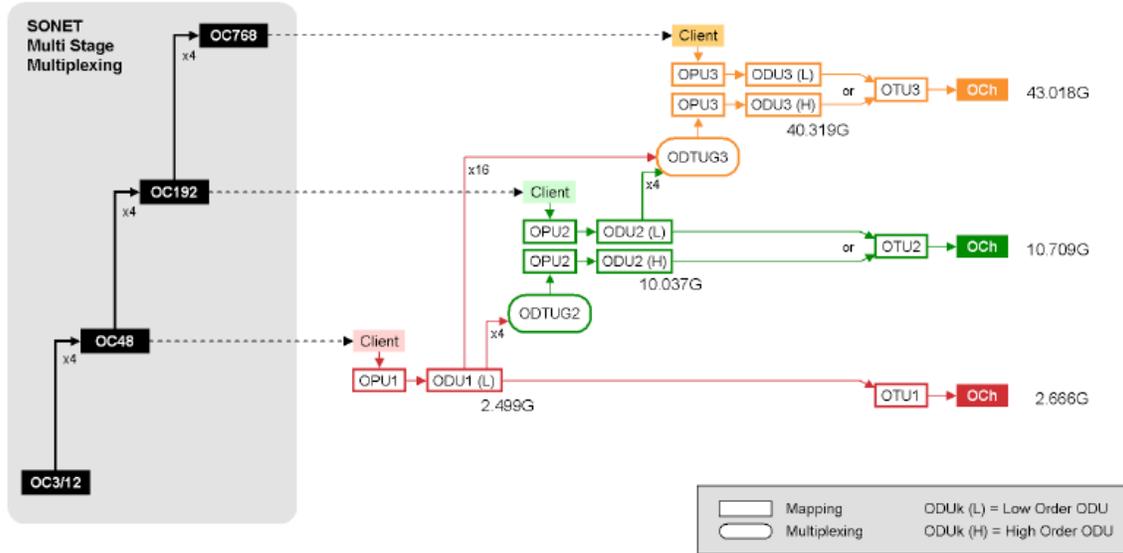
(a) Monitoring the network



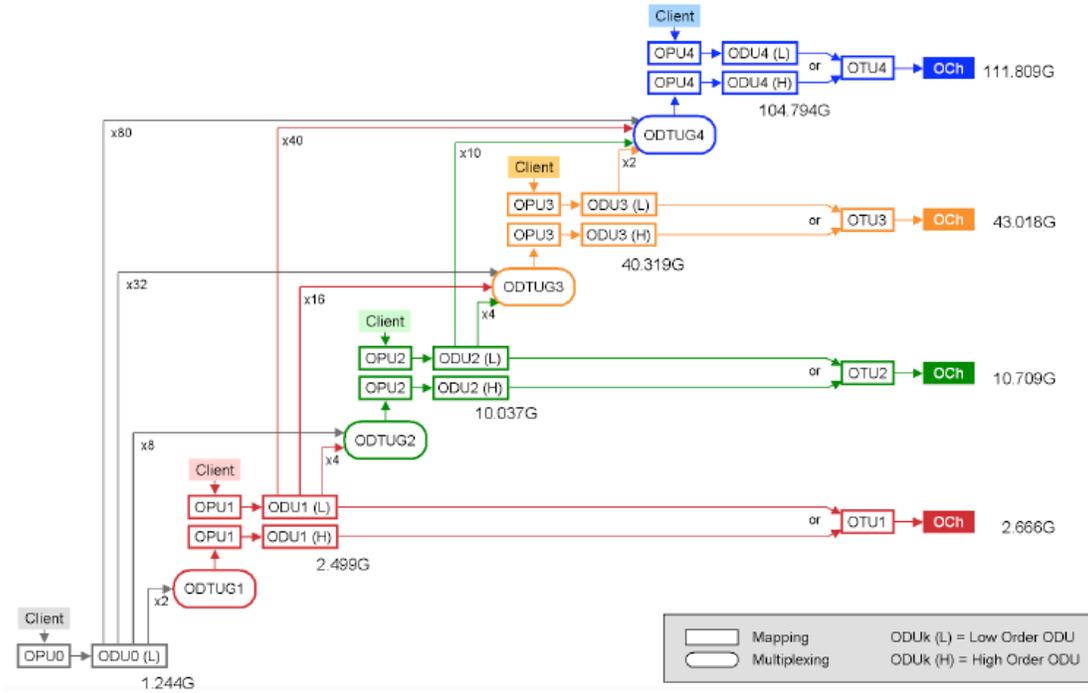
(b) Tandem connection monitoring (TCM)

Figure 2.8: OTN Management Supervision, [BJ10].

2. Background on IP-over-WDM and *p*-Cycle Survivable Networks



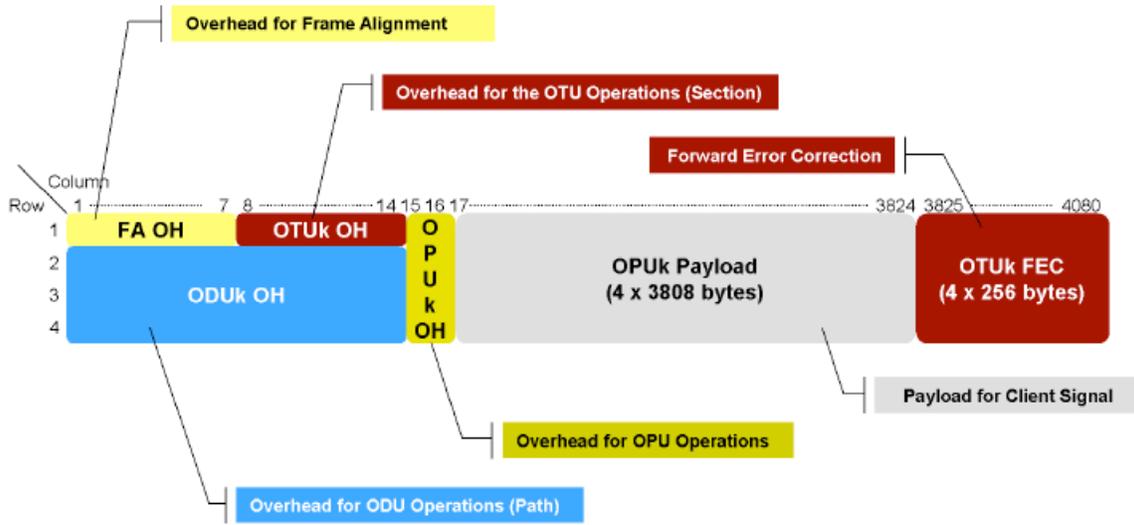
(a) Pre-2009 OTN one stage with SONET mapping



(b) Post-2009

Figure 2.9: OTN Multiplexing Structure, [BJ10].

2. Background on IP-over-WDM and *p*-Cycle Survivable Networks



(a) Frame structure

G.709 Defined Rates

Based on support for GE	ODU0	1.244			
Based on support for OC48/STM16	ODU1	2.499	⇒	OTU1	2.666
Based on support for OC192/STM64	ODU2	10.037	⇒	OTU2	10.709
Based on support for OC768/STM256/40GE	ODU3	40.319	⇒	OTU3	43.018
Based on support for 100GE	ODU4	104.794	⇒	OTU4	111.816

Non-Standard Supplemental Rates

Based on support for 10GE*	ODU1e	10.356	⇒	OTU1e	11.049
Based on support for 10GE**	ODU2e	10.400	⇒	OTU2e	11.095
Based on support for 4 x ODU2e (@2.5G TS)	ODU3e1	41.774	⇒	OTU3e1	44.570
Based on support for 4 x ODU2e (@1.25G TS)	ODU3e2	41.785	⇒	OTU3e2	44.538

* Transparent 10.3125 Gbps Bitstream

** Transparent 10.3125 Gbps Bitstream (Includes Fixed Stuff Bytes)

All rates provided in Gbps

OTU based on RS (255,239) FEC coding

(b) Bitrates

Figure 2.10: OTN Frame Structure and Bitrates, [BJ10].

2. Background on IP-over-WDM and p -Cycle Survivable Networks

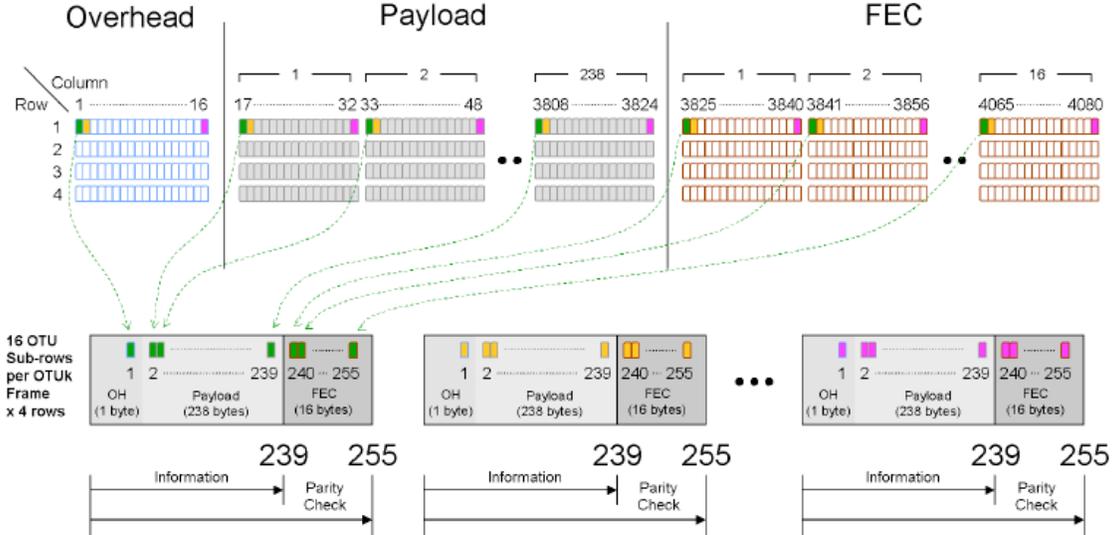


Figure 2.11: Reed-Solomon FEC RS (255/239), [BJ10].

2.2 Survivable Transport Networks

Optical transport networks are very prone to physical failures. Despite physical encasement, the most common failure type remains fiber cable damage arising at a surprisingly high rate from such natural and man-made causes as trench digging, construction work, craftsperson errors, ship anchors, sabotage, tree falls, earthquakes, rodents, fires, floods, and so on. The physical failure of node infrastructures by fire or power loss is certainly unusual, but node outages remain of concern because of frequent crashes and updates related to the software nature of network nodes [EGRG05].

2.2.1 Fundamentals for Survivability in the Transport Layer

For the affected traffic or demands to survive upon failure conditions, survivability measures can be taken at the service or transport levels on the basis of physical route diversity. Generic survivability techniques at the IP/MPLS service level includes adaptive routing within unaffected facilities, demand splitting over multiple routes and application re-attempts. But all of them manage billions of lower-rate connections individually and within *virtual graphs*, and they all require router-oversubscription in order to pre-

2. Background on IP-over-WDM and p -Cycle Survivable Networks

vent congestion at the entrance of actual LSR and IP-routers. The recovery of aggregated transport-level demands, as opposed to multiple lower-rate connections, is more promising and of practical importance because the IP/MPLS service layer will be unaware of wavelength, fiber optic, cable and OXC failures. And the transport layer is equipped with fast switching mechanisms handling huge demand amounts in general.

2.2.1.1 Physical Layer Topology Hypothesis and Requirements

Transport-level survivability is essentially based on the alternate routing of failed demands within the physical layer facilities. For the remainder, the physical graph is defined by two sets of symbols: a finite set of vertices called the *nodes* that are interconnected, each other in pairs, by a set of edges referred to as *spans*.

Graph Topology

This dissertation essentially considers *simple graphs* where one unique span joins each given pair of nodes with no parallel edges or self-loops—as opposed to *multigraphs* that allow parallel spans. Also, test case networks within the thesis are typically *weighted graphs* where a cost, expressed in length-km, is associated with every edge. And in contrast to *directed graphs* in which every span joins an *ordered* pair of nodes, our research mandate is limited to *undirected edges*.

Network Connectivity Enhancements

Highly connected geographical facility routes provide more options for transport-level survivability purposes. The following parameters are essentially used within the thesis to build protection routes and to determine how spare (or conversely, how dense) is a network graph. Two *adjacent nodes* are joined by a span, and the given span is *incident* on the nodes it connects. As a recall, a set of spans that are contiguous on the physical graph define a *route*; while a unit-capacity digital connection between the two end-points of a route is called a *path*. On the other hand, the number of spans incident on a vertex corresponds to the *degree* of the node in question. And the *average nodal degree* of a graph is given by the ratio of twice the number of spans over the number of nodes across the network.

Route Diversity Requirements

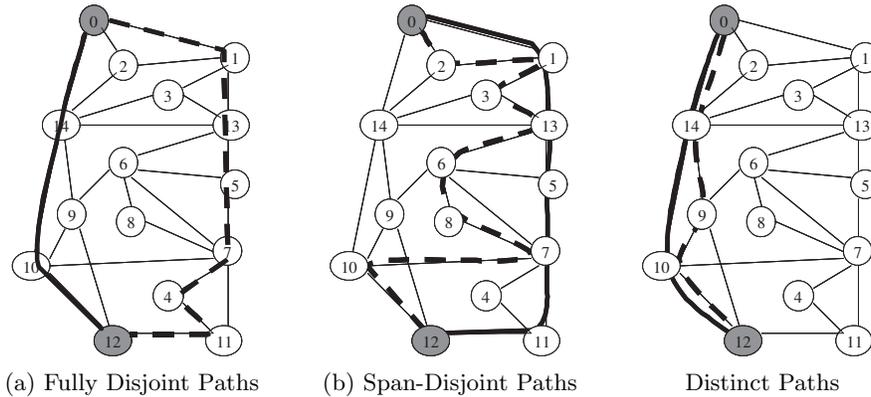


Figure 2.12: *Disjoint and Distinct Routes, adapted from [Gro02a, Gro03b].*

Transport-level survivability seeks to automatically recover failed demands by alternate routing in the physical graph. To make this possible, it is topologically essential to have at least two physically disjoint routes between every pair of nodes. Two *fully disjoint routes* have no node and no span in common; *span-disjoint routes* refer to cases where the disjointness property is limited to spans only. It is not unusual to confuse disjoint routes and *distinct routes* that are different in at least one detail. Reader may want to refer to the illustration in Figure 2.12.

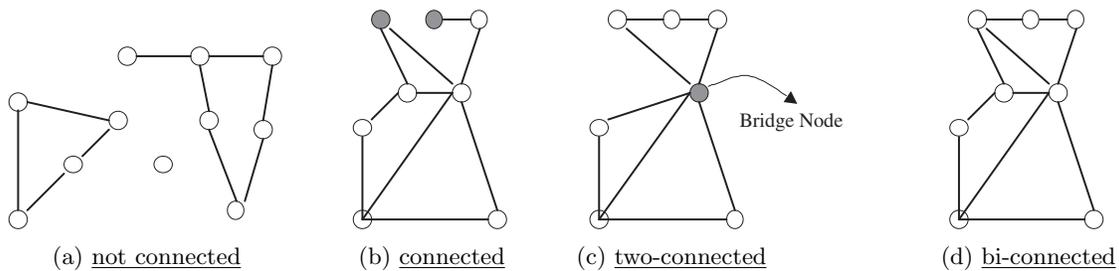


Figure 2.13: *Not-connected, Connected, Two-connected and Bi-connected Graphs, adapted from [Gro02a, Gro03b].*

In a more general manner, a geographical route structure in which there are at least two fully disjoint routes between each node-pair is called a *bi-connected graph*. In contrast, *two-connectedness* implies a minimum of two span-disjoint routes between

2. Background on IP-over-WDM and p -Cycle Survivable Networks

all node-pairs, but there may be a node in common between the two routes; this sometimes results in one or more articulation points referred to as *bridge nodes*. Figure 2.13 shows the subtle difference between not-connected, connected, two-connected and bi-connected graphs.

2.2.1.2 Working versus Spare Capacity Channels

From this point, unless otherwise indicated, all references to traffic, flows, demands, and so on, refer to transport-level demands. The net effect is ultimately that either through IP-over-WDM or through a stack of DS-3, ATM and SONET layers, a set of user services are mapped onto a set of physical high-capacity transmission, multiplexing and signal switching facilities that provide transmission paths to support the logical connectivity and capacity requirements of all service flows. The literature provides many advanced techniques for routing purposes, but these are out of the scope of this thesis.

Herein, we simply assume shortest hop-count or distance weighted routes pre-determined by any basic routing algorithm such as Dijkstra, *k-shortest path* or *depth-first-search* (DFS). And those routing operations are completely separate from the protection of resulting working paths. As a result, every span within the transport-level infrastructure is capacitated with two sets of channels, expressed in terms of wavelenghts, OC-n, STS-n, etc. The first set encodes *working capacity channels* that are in service under no failure conditions, as part of working paths bearing one demand-unit each. On the other hand, *spare capacity channels* are available for use in the event of failure; so they remain idle or serve for low priority demands under normal network conditions. The set of all spare channels form a *reserve graph* for efficient response to network disruptions. The literature proposes numerous transport-level survivability schemes aimed at reducing reserve network costs while automatically providing the intended capability.

2.2.1.3 Measures of Survivability

One typically distinguishes between three measures of survivability: reliability, availability and restorability. Reliability level indicates the probability that a system operates without a service-affecting failure for a given amount of time. Availability level corresponds to the

2. Background on IP-over-WDM and p -Cycle Survivable Networks

probability that a continuously operating system undergoing repair after any failure is in its normal state at any random time in the future.

Restorability is the only measure of survivability considered within the thesis. It gives the fraction of working demand flows affected by a failure that are recovered or for which a recovery path set solution is feasible. In the thesis' series of research studies, the first requirement for reserve backbones is full protection against single span failures; this is denoted $R_{1\text{-span}} = 1$ and the transport network is said to be 100% span restorable. Such symbols as $R_{1\text{-node}}$ and $R_{2\text{-span}}$ then denote restorability levels from node and dual span failure perspectives, which are of secondary concern.

Redundancy is the ratio of spare capacity required in the transport network to meet restorability goals to working capacity required only to route demands without survivability concerns. Network redundant capacity is widely used as a cost metric, typically to simplify network planning. But real-world network costs correlate with capital expenditure (CapEx) and operating/revenue expenditure (OpEx). Most of the time, the thesis' studies also limit to capacity channel-kms considerations; but Chapter 4 addresses how this influences or connects to real-world CapEx costs.

The Figure of “50 ms” — Restoration times constitute another key performance evaluation criteria for survivability schemes. A techno-cultural goal of 50 ms is usually mentioned in reference to earlier transmission systems that typically required 20 ms for fault detection, 10 ms for signaling and 10 ms for switching operations, with an allowed margin of 10 ms. Faster restoration is certainly desirable, but restoration goals must be carefully set in conjunction with costs that may be paid by limiting the available choice of network architectures. The entire scope of transport-level survivability schemes becomes available with something more like 200 ms recovery time requirement, which is not an issue of real importance for many client-services.

2.2.2 Transmission System Layer Survivability Schemes

Transmission system layer survivability relies on essentially static protection structures, typically based on the whole-fiber or large waveband level of granularity, where the alternate path taken by every single demand unit in the event of failure is clearly known in

2. Background on IP-over-WDM and p -Cycle Survivable Networks

advance. Once installed and tested, the operation system of survivability schemes is relatively simple and self-contained, and the restoration process is quite fast (usually under 50 ms). Historical examples of transmission level survivability schemes mainly include automatic protection switching (APS) systems, unidirectional path-switched rings (UPSRs) and bidirectional line switched rings (BLSRs) widespread in SONET networks.

2.2.2.1 Automatic Protection Switching (APS)

The basic idea behind automatic protection switching (APS) is to provide a standby transmission resource that is kept in full operating condition and used to replace any of the other traffic. Two main functions characterize the operation of APS systems: the *head-end-bridging* and the *tail-end-transfer*. The tail-end node detects failures on the working system and signals this to the head-end node, which responds by switching traveling demands onto the bridge. There are three main variants: i.e. 1+1 APS, 1:1 APS and 1:N APS.

In 1+1 APS, the same signal is sent on both the working channel and on the spare channel. The receiver monitors both signal copies and switches from one to the other if either fails. As the head-end-bridge is always established and requests no specific action from the tail-end node in the event of failure, 1+1 APS provides the fastest possible switching speed in the domain of network survivability. In 1:1 APS, the head-end-bridge is established in response to the event of failure. The signal is sent on the working channel alone under normal network states, and is moved onto the spare channel upon failure of the working channel. This presents the advantage of handling low-priority demands over the spare channel under normal network states; but head-end-bridge establishment also makes 1:1 APS slightly slower than 1+1 APS because recovery times now include failure detection, signaling and switching operations.

But 1+1 APS and 1:1 APS denote a *dedicated* standby arrangement, meaning that each working channel/system has its own backup. Compared to the dedicated protection mode, *shared protection* is generally preferable because it gives rise to significant spare capacity savings. An APS system in which multiple working channels share one single standby protection system is denoted 1:N APS. If one of the working channel fails, the tail-end-node detects the failure, checks the spare channel availability, and requests a head-end-bridge

2. Background on IP-over-WDM and p -Cycle Survivable Networks

if the protection channel is available. When the failed channel is repaired, the traffic is sent back onto the working channel; this is called *reversion*. The overall principle is also amenable to $k:N$ APS ($1 < k < N$), where k standby channels protect N working channels.

The original intent and ability of APS systems was only to recover from single channel or fiber failures. Cable cuts were not covered at all because both working and standby channels/fibers were part of the same geographical routes. To withstand entire span failures, 1+1 DP APS and 1:1 DP APS extend pure 1+1 APS and 1:1 APS systems in a way that any protection path is physically disjoint from the working channel/path it handles; DP stands for *diverse protection*. But there is no such ability in 1:N APS systems.

2.2.2.2 SONET Survivable Rings

Ring-based protection schemes are enhancements of 1+1 APS and 1:1 APS technologies. As a simple extension to the SONET APS signaling protocol, they are perceived as easy to understand, develop and operate. But unlike APS systems, survivable rings address both the need for single-channel and span failure protection, and even protect against node failures as well. Moreover, a ring topology collects all demands together to exploit the *economies-of-scale* in transmission technology (e.g. an OC-192 ring is a lot less expensive than four OC-48). A survivable ring can support either unidirectional or bidirectional routing, and performs protection switching either at the path level or at the multiplexed line level. UPSR and BLSR are two main types of SONET rings.

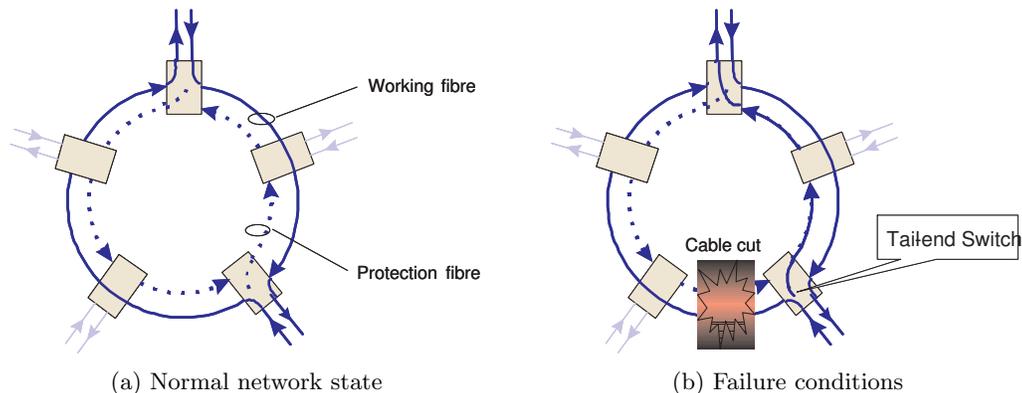


Figure 2.14: Operational Principle of UPSR, adapted from [Gro02a, Gro03b].

2. Background on IP-over-WDM and p -Cycle Survivable Networks

UPSR is inherently a two-fiber structure that uses unidirectional routing and path-level protection. As shown in Figure 2.14, the two fibers are running in opposite directions; one of them can be considered for no failure states while the other is used under failure circumstances. In normal operation, all working demand flows are routed in one single direction over the same fiber. And in the restoration mode, each receiver selects an alternate end-to-end path through the other ring, regardless of where the actual failure occurred. Switching on one channel has no effect on other channels, meaning that UPSR switching decisions are independent on a tributary-by-tributary basis. Also, there is no need for reversion after the failure is repaired. UPSR is primarily used in access applications where distances are not great, but it is quite expensive for the following reasons. First, the unidirectional routing implies that in any cross section of the ring, we would find one unidirectional instance of every demand flow between nodes of the ring because every bidirectional demand relation circumnavigates the entire ring. On the other hand, the protection ring must have a spare capacity greater or equal to the sum of all the bidirectional demand quantities between nodes of the ring. However, UPSR is as efficient as BLSR (to follow) under pure “hubbed” demand patterns.

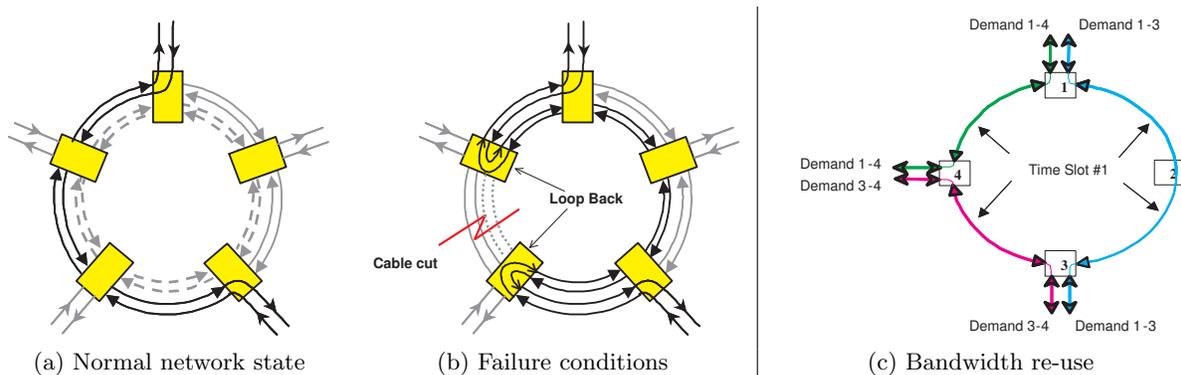


Figure 2.15: *Operational Principle of BLSR and Illustration for Possible Bandwidth Re-use, adapted from [Gro02a, Gro03b].*

BLSR operating principle stands in contrast to that of UPSR because the former uses bidirectional routing and performs a line-level loopback protection. The principle is defined for both two-fiber and four-fiber variants: a two-fiber BLSR comprises two fiber optics running in opposite directions, each fiber having its channels arranged in two groups; one

2. Background on IP-over-WDM and p -Cycle Survivable Networks

for normal routing and the other for protection. A four-fiber BLSR has two working and two protection fibers, each group running in opposite directions. Figure 2.15(a) shows that in both variants, any bidirectional demand unit travels in opposite directions over the same route through the ring under normal network state. In Figure 2.15(c), this gives opportunity for *bandwidth re-use* given that demand relations can be routed over shortest paths within the ring and do not cross all spans across the ring. In restoration, the composite optical line transmission signal is switched to the other direction around the ring, specifically around the failure on the other channel groups or fiber-pair—as shown in Figure 2.15(b). To offer the capability, the BLSR capacity must be greater or equal to the largest sum of demands routed over any one span on the ring; further, the protection capacity must equal the largest working capacity cross-section of any span on the ring.

As an extension of the APS technology, SONET rings provide a straightforward failure detection, signaling and bridging, with total switching times typically under 50 ms. But the apparent operational simplicity when considering one single ring structure is actually extremely complex to design and develop, especially in practical networks comprising multiple interconnected rings. Survivable rings also impose a normal state routing within the ring, which is inflexible to changes in the demand pattern. From both working and spare capacity perspectives, survivable rings are quite inefficient in overall capacity usage. These are some reasons for the growing interest in the mesh-based alternatives that follow.

2.2.3 Logical Layer Mesh-Type Survivability Schemes

Despite fast restoration times and apparent simplicity, transmission layer survivability schemes generally require high installed capacity for demand-served and are really hard to accommodate with demand growth and multiple service classes. APS systems that recover from span failures use dedicated protection, while UPSR and BLSR require a ring-constrained routing and manage fiber-level protection structures. But the DWDM technology allows for the manipulation of single wavelength channels or wavebands. And OXCs can continually self-organize the mapping of physical transmission to logical transport configuration in order to suit time-and-spatially varying demand patterns. So with the advent of DWDM and OXC technologies, it is possible to take advantage of the highly mesh-like topology of most actual networks.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

Logical layer survivability schemes are generally mesh-oriented, with features not possible in APS and SONET ring systems. Specifically, mesh-based routing is quite flexible and adaptable to unforeseen patterns of demands under much lower working capacity requirements. As well, the generalized re-routing over a physically diverse graph can permit greater sharing of spare capacity, with redundancy going down in proportion to the network average nodal degree. Moreover, the network can now be its own computer for real-time solutions of the re-routing problem, without any external control or databases. This certainly increases recovery times but as earlier stated, a time of 50 ms is not necessary for most access technologies. The literature proposes a panoply of logical layer mesh-type survivability schemes that can be classified in many different ways: i.e. shared versus dedicated protection, protection versus restoration schemes, span versus path-oriented paradigms, ring versus mesh-based architectures, etc.

2.2.3.1 Shared versus Dedicated Protection

Networks are inherently mesh-like topologies. The logical cross-connect layer allows distributed mesh-protection that exploits network connectivity to permit sharing of redundancy. In *shared protection/restoration*, a given working path may have multiple protection paths for different failure scenarios, each protection path following a distinct route; many protection paths for span-disjoint working paths have several spare channels in common; and the share capacity on each span contributes to restorability of many spans. This stands in contrast to dedicated protection schemes, such as 1+1 APS and 1:1 APS, where one protection path is pre-established between the end-nodes of every single working path. From total spare capacity requirement perspectives, dedicated protection typically involves at least 100% redundancy where shared protection is well under 100%.

2.2.3.2 Protection versus Restoration

Both protection and restoration techniques just boil down to the technique of how spare capacity is shared or dedicated among demands [DC09]. So as far as this researcher is concerned, either protection or restoration can be used as synonyms to encompass survivability schemes, spare capacity requirements, and recovery processes and speeds. The

2. Background on IP-over-WDM and p -Cycle Survivable Networks

present dissertation follows the above line of reasoning; but we recognize that many people tend to make a subtle distinction between protection and restoration terms.

- If all replacement paths are known in advance, are pre-cross-connected and need only be accessed at end-points in the event of failure, the survivability scheme is referred to as a *protection* technique. With the argument of full pre-cross-connection, protection schemes provide the fastest possible recovery times in the domain of network survivability.
- *Restoration* implies that alternate routes are found adaptively based on the failure and the state of the network at the time of failure, and cross-connections to assemble the replacement paths are also made in real-time. This allows great spare capacity savings but restoration times are much longer than with protection techniques.
- A tradeoff is possible with *pre-planned restoration*. Replacement routes are known in advance and cross-connection maps for fast local action are in place at all nodes but cross-connection is required to assemble the restoration path-set in real-time upon failure.

2.2.3.3 Demand-wise Shared Protection (DSP)

APS, UPSR, BLSR and all system layer survivability schemes are inherently of the protection class because standby capacity is pre-defined and self-contained in the transmission system. In contrast, logical layer protection schemes are essentially recent survivable network architectures. A typical example is demand-wise shared protection (DSP) of which operating principle is inspired by $k:N$ APS systems. Given a demand relation between a pair of nodes, a set of two or more disjoint routes is identified between the nodes; one of those routes is set as protection route and the demand units are divided as equally as possible among the remaining disjoint routes, so that spare capacity placed upon the protection route is the largest number of lightpaths placed on any single working route. DSP literature claims a capacity efficiency comparable to that of 1+1 APS, and an 1:1 APS-like restoration time [KZJH05, WOZ⁺05, GKO⁺05, HJK⁺06, FG07, For09].

2.2.3.4 Shared Backup Path Protection (SBPP)

Shared backup path protection (SBPP) is another extension of APS systems that can be implemented either in the logical cross-connect layer or in the IP/MPLS service layer.

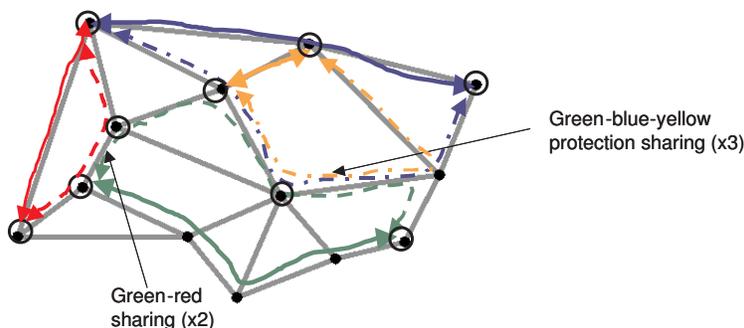


Figure 2.16: *Illustrating Shared Backup Path Protection, with solid lines representing working paths while colour-matching dashed lines are for replacement paths, adapted from [Gro02a, Gro03b].*

SBPP is part of the pre-planned restoration class in the sense that each working path has a fixed disjoint backup route pre-determined at the path-provisioning time; when a protection path is needed, it is formed by cross-connecting spare channels along the backup route. This is like 1+1 APS, except that spare capacity can be shared among disjoint working paths as shown in Figure 2.16.

2.2.3.5 Path Restoration Scheme

In contrast to transmission systems, logical layer survivability schemes can be either protection, restoration or pre-planned restoration. A typical example of logical layer restoration technique is *path restoration* that consists of abandoning damaged pre-failure paths entirely, making previously used (but now unnecessary) working capacity available as additional spare capacity, and rapidly re-provisioning new paths end-to-end. To illustrate, Figure 2.17(a) shows a three working paths severed by a cable cut. In 2.17(b), the failure is detected and surviving portions of all affected working paths (i.e. the stubs) are released; capacity that previously supported working paths is now deemed available as spare capacity upon restoration. So end-to-end replacement paths are established using both shared spare capacity and released working capacity as in 2.17(c).

With *stub release*, path restoration is theoretically the most efficient possible scheme from capacity design and multiple failure consideration standpoints. But by reducing the amount of spare capacity needed in the network, stub release greatly complicates things from the operational viewpoint. More specifically, an automatic signaling protocol such

2. Background on IP-over-WDM and p -Cycle Survivable Networks

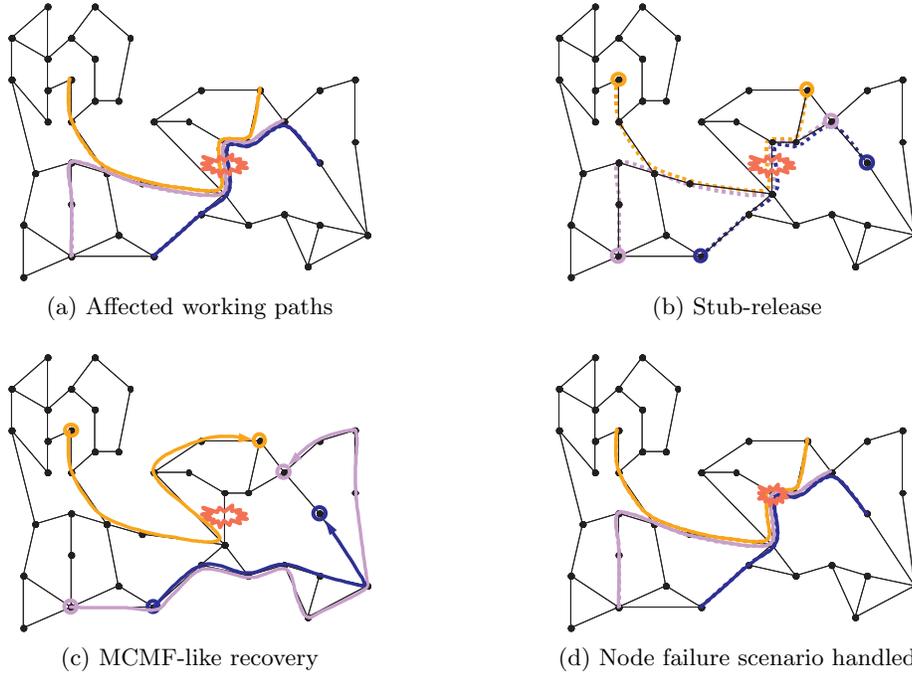


Figure 2.17: *Illustrating Path Restoration, adapted from [Dou06].*

as *alarm inhibit signal* (AIS) is required to rapidly release the surviving working stub capacities. The capacity design and path finding problems for path restoration follow the multi-commodity maximum flow (MCMF) optimization problem, which is considerably complex and time-consuming in real-time. And after physical repair, it requires an even more complex reversion.

2.2.3.6 Span versus Path-oriented Paradigms

APS systems, UPSR and BLSR rings, DSP, SBPP and path restoration all provide end-to-end replacement of working paths affected by a failure. Given a specific working path, the same end-nodes activated reaction occurs regardless of where failure strikes the path in question. As shown in Figure 2.17(d), this gives the opportunity to recover from intermediate node failures along the path; the only requirement will be to extend the span-disjointness property to full disjointness between any set of working paths sharing spare capacity along their replacement paths, and between any working route and its backup

2. Background on IP-over-WDM and p -Cycle Survivable Networks

routes. Path-oriented survivability is well suited for router-centric control where the network core is “dumb” while the edge is “smart.” But this will require rapid fault detection and backup activation, which imply high dependency on conventional software, databases and current ideas of Internet-like global state dissemination, etc.

Span-based architectures stand in contrast to the path-oriented paradigm. As shown in Figure 2.18, a set of working paths affected by a span cut is restored by a set of *local replacement paths between the end-nodes of the failed span*. This guarantees straightforward failure detection and reversion where path-oriented architectures require advanced signaling to offer the capability. But span protection/restoration a priori provide no node protection capability, and (end-to-end) restored state paths may possibly have some loopback. From the spare capacity requirement perspective, replacement paths are routed over spare capacity distributed in a general fashion through the surviving portion of the network. This sharing efficiency increases with the network connectivity, i.e. average nodal degree, but remains limited in comparison with stub release feature. From the network design viewpoint, finding a suitable restoration path set is equivalent to a single commodity maximum flow (SCMF) problem or a k-shortest path routing between failure end-nodes, which are much simpler than the MCMF problem for path-oriented architectures. In span-protecting architectures, replacement paths are pre-determined before any failure; while in span restoration the SCMF problem is solved in real-time after the failure occurred.

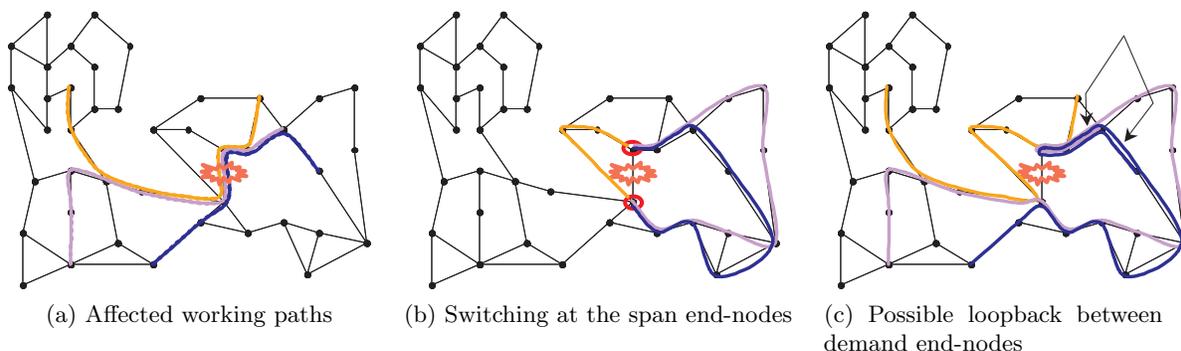


Figure 2.18: *Illustrating Span Protection/Restoration, adapted from [Dou06].*

2.2.4 Pre-cross-connected Protection Structures

There is no universal winner among the traditional survivability schemes above. Each has both quantitative and qualitative pros and cons, with regards to two mutually exclusive trends. APS and ring-based systems are characterized by 50 ms restoration times, simple switching operations and low-cost ADMs. But the penalty for all of those features is the complex network planning, the high installed capacity incurred for the demand served, and the difficulty of accommodating network changes (e.g. topology, demand growth) and multiple classes of protection. On the other hand, mesh-type network architectures allow more flexibility in the routing of demand flows for multiple protection classes; and they provide easy and efficient design solutions, well under 100% redundancy. However, they also incur slow switching times that may reach up to 1.5 s with pure restoration schemes, and they require intelligent and relatively expensive DCS/OXCs.

DWDM potential for *pre-cross-connection* network architectures opens a totally unexpected way to gain advantage of both system layer protection schemes and mesh-based survivability methods. Chains of pre-cross-connected capacity may coalesce into protection structures such as SONET systems (i.e. APS, UPSR and BLSR) to guarantee fast restoration times, straightforward backup activation and low-cost technology. But pre-cross-connected protection structures also continue to operate on fully-meshed facilities, which allow more flexibility in the routing of demand flows and a shared pool of spare capacity for multiple protection classes, and which provide easy and efficient network design solutions. Unlike with physically pre-connected architectures and pure mesh survivability schemes, the penalty for low-cost network elements and fast restoration times is no longer high-capacity requirements and complex design (and vice versa). The literature differentiates between tree, trail and cycle-like pre-cross-connected structures.

2.2.4.1 p -Tree Protection Structures

[GCM⁺03, SHY04] discuss the fundamentals of the p -tree survivability technique, which strictly uses tree-like structures to provide the capability. For example, the blue lines in Figures 2.19(a)-(b) represent a p -tree formed out of a number of spare channels pre-cross-connected to each other at the network nodes. In the restoration mode, one distinguishes between span and path-protecting p -trees. The span-protecting p -tree in 2.19(a) protects

2. Background on IP-over-WDM and p -Cycle Survivable Networks

one channel on every span that has both its end-nodes but is not itself on the p -tree structure, meaning that on-structure spans of any given p -tree must be protected by complementary p -trees. In contrast, the path-protecting p -tree in 2.19(b) protects individual working paths that have both their end-nodes on the p -tree structure, and for which the p -tree contains a path through itself between those end-nodes that is disjoint from the failed working path and the protection path in question. With both span and path-protecting p -trees, it is easy to see that many spare channels in the p -tree remain unused during the restoration process and a given p -tree can provide at most one protection path to a single working route. But this gives the possibility for multiple spans or routes to be protected by two disjoint protection paths on the same p -tree simultaneously.

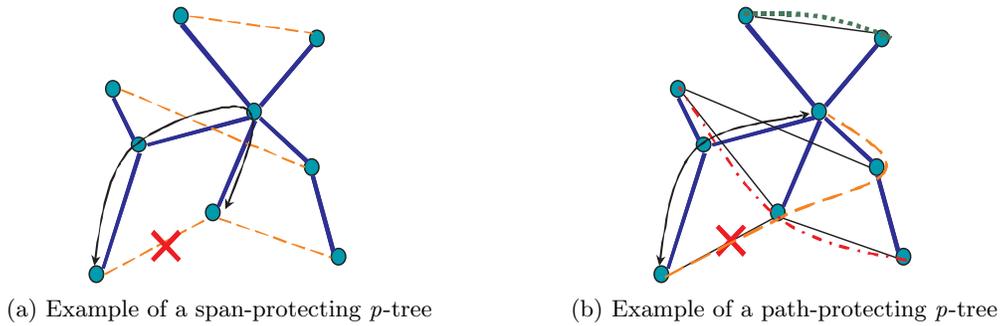


Figure 2.19: Operating Principle of Span and Path-Protecting p -Trees (in blue), adapted from [GFO⁺06].

2.2.4.2 Pre-Cross-Connected Trail (PXT) Linear Structures

The pre-cross-connected trail (PXT) survivability scheme in [CCF04] uses linear-like structures of degree 2, which are much simpler than p -tree multiple-degree pre-connections (assumed possible regardless of whether or not this theoretical construction has any real-life meaning). Figure 2.20(a) gives an example of PXT. In 2.20(b)-(c), the PXT protects on-PXT and off-PXT working paths that have end-nodes on the PXT structure and are disjoint from their protection path formed through the PXT. The scenario in 2.20(d) shows that disjointness between working and protection paths is essential to provide the capability.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

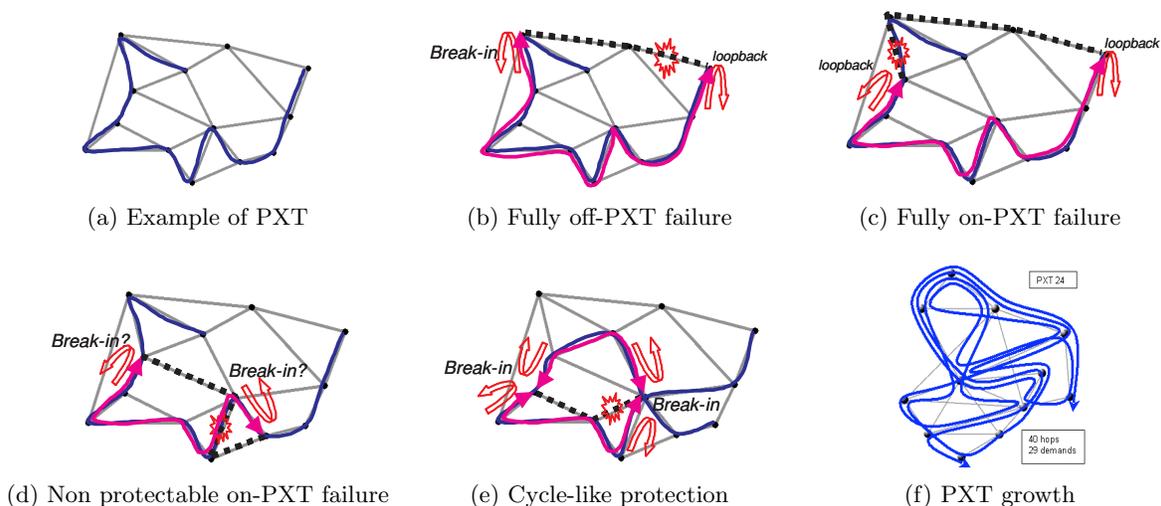


Figure 2.20: *Illustrating PXT Operating Principle, adapted from [Gro03b, Gro02a, GFO⁺06].*

2.2.4.3 Narrowness of Tree and Trail-like Geometric Shapes

But either p -trees or PXTs are themselves greedy structures that may cross the same nodes and/or spans more than once. This provides some interesting protection possibilities, as shown in Figure 2.20(e); but the protection structure can also become incredibly complicated as in Figure 2.20(f). One can only consider *simple* structures not allowed to self-intersect, but the set of demands protected by such a more restricted p -tree or PXT would also be very much smaller than that of a standard p -tree/PXT. Nevertheless, [GGC⁺07, GG06, GG07c, Gru09] mentioned several issues incurred with p -tree and PXT related networks: i.e. the fact a working path has a viable protection path through PXTs for every possible span failure but not necessarily the same for every specific failure. Thereby, they require failure-specificity actions for proper restoration, the possible signal multiplexing issue for tree-connected nodes with two or more branches meeting in the incoming direction, and more significantly the complexity of p -tree and PXT network designs. The following survivability scheme uses the cycle geometric shape to overcome all of those issues.

2.3 Pre-configured Pre-cross-connected Protection Cycles

p -Cycles constitute a tremendous survivability technique that unifies all of the other protection and restoration mechanisms from many different aspects. Like the p -tree and PXT structures, p -cycles keep the pre-cross-connection property which may be key to use in transparent optical networks; and the original intention with p -cycles was to combine the operational simplicity, low-cost network elements and fast restoration times of ring systems with the flexibility and freedom of meshed topologies in the routing of working paths. [GG07b] provides a unified framework for comparing p -cycle and p -tree capabilities. But p -cycles use the cyclical geometric shape that is much more manageable than the p -tree and PXT greedy structures, and supports very well optimization-based methods allowing more capacity efficiency in the design. Moreover, p -cycles can be implemented at any level of the transport networking infrastructure. In the transmission system layer, p -cycles require a minor variation on SONET ADMs in order to extend BLSR protection capabilities for more capacity efficiency; with the SONET ring equipment widely deployed in the mid-90s before the advent of DWDM systems, p -cycles make an easy transition from SONET systems to mesh-type architectures. But in regards to the mesh-like efficiency, p -cycles are usually considered as a logical cross-connect survivability scheme. They can be configured either in a static or adaptive way. Conventionally, they follow the span-protecting paradigm with such advantages as locality of switching actions, minimal database state dependencies, high speeds and span-like working envelope for demand uncertainty; but unlike other span-oriented survivability schemes, span-protecting p -cycles have an intrinsic ability to also recover from node failures. As other p -tree and PXT pre-cross-connected structures, the basic operating principle of p -cycles is extendable to path-segment protection and end-to-end path protection. MPLS labels may be used to create virtual p -cycles in the IP-service layer, which is prone to frequent IP router outages due to their software-related nature. We now give details on the p -cycle survivability technique.

2.3.1 Basic Operating Principle, Ring and Mesh-like Benefits

[GS98a] named p -cycles in reference to **pre**-configured, **pre**-connected **pro**tection cycles. So a p -cycle can be defined as a cyclic pre-cross-connected closed path formed out of the spare

2. Background on IP-over-WDM and p -Cycle Survivable Networks

capacity. Conventionally, a p -cycle operates as a span protection scheme; it provides one protection path to every on-cycle span and two protection paths to any straddling span, which has both its end-nodes but is not itself on the cycle structure. Figure 2.21 illustrates the p -cycle restoration principle—2.21(a) gives an example of a p -cycle; in 2.21(b), the p -cycle reaction to an on-cycle span failure is logically identical to a unit-capacity BLSR loopback through the surviving side of the ring; and in 2.21(c), the p -cycle reaction to a straddling span failure is to break-in failed signals onto the two sides of the cycle circumference.

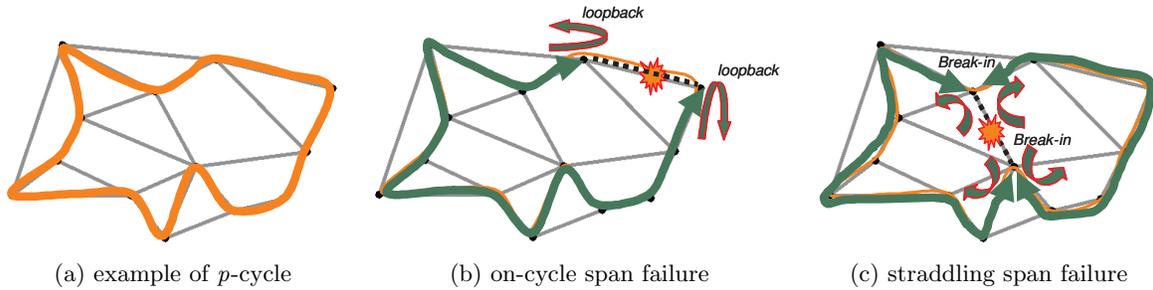


Figure 2.21: *Operating Principle of Span-Protecting p -Cycles, adapted from [Gro03b, Gro02a].*

The p -cycle operational principle above is based on a minor variation of ring behavior under failure circumstances; but the extension of protection capabilities to straddling spans and the full mesh-routed conditions make a huge difference on capacity requirements. Actually, working path routing neither conforms to ring systems nor limits inter-ring transfer points. Rather, the routing and provisioning of working paths is proceeded without regards to p -cycle structures formed only in the sparing layer. So working paths may freely go via shortest routes over the entire facilities graph for more efficiency. On the other hand, ring-like attributes contribute for one protection path per use to the restoration of any span on the p -cycle structure; and in addition, the same p -cycle yields up to two restoration paths to each straddling span. The significance of straddling failure recovery can be illustrated by the example in Figure 2.22, where a p -cycle consumes 13 unit-hops of spare capacity and protects one working signal on the 13 working spans and two working signals on the 9 straddling spans for a total redundancy of only 42%. In a more general manner, a fully-loaded *Hamiltonian* p -cycle that crosses once all N nodes across the graph reaches the redundancy limit $[1/(N - 1)]$. And any p -cycle of N circumference-hops may have up to

2. Background on IP-over-WDM and p -Cycle Survivable Networks

$[N(N-1)/2 - N]$ straddling span relationships, each straddling span bearing two working channels and exactly zero spare channels. So p -cycles approximate the investment of span restorable mesh in spare capacity, which is well under 100%.

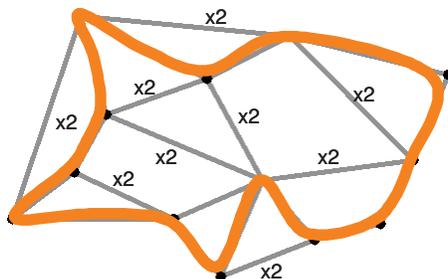


Figure 2.22: *Impact of Straddling Failure Restoration on the Redundancy, adapted from [Gro03b, Gro02a].*

But p -cycles are just ring-like structures, thereby keeping many important features of SONET rings. More specifically, p -cycle restoration times have BLSR-like speed although p -cycles run on mesh-type infrastructures. Fast restoration times are possible because protection capacity is fully pre-cross-connected and switching actions are well known prior to failures. And there is no specific signaling requirement to locate failures or to activate proper backup processes; but as a span-protecting scheme, p -cycles perform a straightforward failure detection at the two neighbor nodes and only those nodes do any real-time switching for restoration. With BLSR-like attributes, span-protecting p -cycles inherently recover on-cycle flows transiting through a failed node through a loopback at the nearest neighbor nodes; this property stands in contrast to the common knowledge that span-protecting architectures provide no protection against node failures. As will be seen later, actual low-cost ADM-like elements support p -cycles well; so from the ring-to-mesh evolution perspective, p -cycles may constitute a promising ring-mesh hybrid scheme.

2.3.2 Whole p -Cycle Network Design

A significant part of the p -cycle literature is dedicated to network design questions: e.g. optimal spare capacity design, joint working and spare capacity optimization, transparent reach limitation in the restored network state, modular capacity (as opposed to unimodularity), node restorability maximization, multiple prioritized protection policies, availability-

2. Background on IP-over-WDM and p -Cycle Survivable Networks

managed network design, dual-failure restorability enhanced, etc. But all of those concerns are based on minor variations or extensions of the minimum spare capacity design problem.

2.3.2.1 Conventional p -Cycle Minimum Spare Capacity Design Problem

The conventional p -cycle minimum spare capacity design assumes a simple network graph and a set of (*say*) transport-level demands flowing within the physical graph in the normal network state. The goal pursued is to protect those working demand flows by selecting an effective set of cycle structures among all candidates, and capacitating them at the least possible channel-km requirements for 100% restorability against single span failures. To illustrate, Figure 2.23(a) shows an unprotected network with working capacities atop spans, Figure 2.23(b) proposes a designed set of p -cycles for use under failure conditions and Figure 2.23(c) derives the spare capacity required on the edges to accommodate the design—Appendix A demonstrates the effectiveness of this solution. From another perspective, the graph in 2.23 contains 135 distinct cycle structures, but only five of them appear in the final solution. It requires optimization methods to identify p -cycles of the highest *collective* merit.

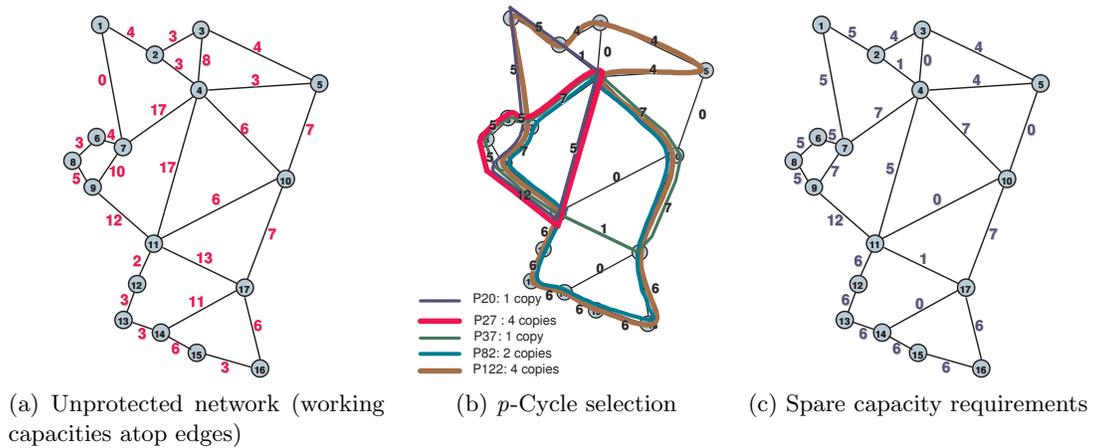


Figure 2.23: *An Example of a Whole p -Cycle Design.*

2.3.2.2 Integer Linear Programming (ILP) Model for Conventional p -Cycle Minimum Spare Capacity Design

Integer linear programming (ILP) is a mathematical model for planning (programming) decisions that optimize a linear objective function and satisfy limitations imposed by linear mathematical constraint functions, with all decision variables restricted to integer values. Following the generic form of ILP models, ILP₁ proposes a mathematical formulation for the conventional p -cycle minimum spare capacity design problem. The following definitions serve for ILP₁ mathematical formulation.

Sets Two “sets” of data are defined to record spans and cyclical structures available within the network.

- S is the set of spans in the network, indexed by i for failing spans and j for surviving spans or spans in general.
- P is the set of candidate cycles, determined by a pre-processing method and indexed by p .

Input Parameters indicate the cost for placing a unit of capacity on each edge, protection-relationships between spans and cycles, and working capacities to be protected on edges.

- C_j is the cost of each unit of capacity (i.e. channel) on span j .
- w_i is the number of working channels to be protected on span i . This is an input arising from whatever routing process is applied to demand matrix.
- $x_i^p \in \{0, 1, 2\}$ encodes the number of restoration segments that one copy of p -cycle p may provide to span i ($x_i^p = 2$ if i straddles p , $x_i^p = 1$ if p crosses i , $x_i^p = 0$ otherwise.)

Decision Variables pertain to cycle selection and the spare capacity required to build p -cycles involved in the final solution.

- s_j is the number of spare channels on span j , in the design.
- η^p is the number of unit-sized copies of p -cycle p , in the design.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

ILP₁ Mathematical Formulation — Equations (2.1)-(2.3) comprise ILP₁ for conventional p -cycle minimum spare capacity design. The objective function (2.1) is used to minimize total spare capacity requirements. The use of constraint (2.2) guarantees full protection in the event of single span failures, through the p -cycles built within span spare capacities in equation (2.3).

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j. \quad (2.1)$$

subject to:

$$w_i \leq \sum_{p \in P} x_i^p \cdot \eta^p, \quad \forall i \in S. \quad (2.2)$$

$$s_j = \sum_{p \in P: x_j^p = 1} \eta^p, \quad \forall j \in S. \quad (2.3)$$

2.3.2.3 Depth-First-Search (DFS) Algorithm for Candidate Cycle Enumeration

ILP₁ for conventional p -cycle minimum spare capacity design requires the pre-enumeration of all possible candidate cycles available within the network instance under consideration. [MD76, Joh75] discuss many routing algorithms that may be helpful for this purpose, but we will limit ourselves to the depth-first-search (DFS) all-cycle finding algorithm for which [Gro03b] claims a general use of in network planning. According to his pseudo-code, on pp. 220–221, the DFS procedure to find a cycle starts at an arbitrary node s and is looking for a path $(s, v_1, v_2, \dots, v_k)$. Vertices v_j are marked as added to the path; and the search of new nodes for the current path stops when the path is blocked by a marked node $v_{k+1} \neq s$, when the circumference-size limit of a potential cycle is exceeded, or when a cycle is found (i.e. $v_{k+1} = s$). After all edges from a node v_j have been explored, v_j is unmarked if a cycle was found or if the given cycle circumference limit was exceeded during the current depth search. Recursively, any other marked node is unmarked if it is adjacent to v_j and not in the current path, and so on for vertices adjacent to the current unmarking node. After that, the DFS search backs up to the previous node v_{j-1} and the cycle finding procedure continues until backing up to the root node s .

At this point, all cycles traversing the explored edge (s, v_1) have been found; so the edge in question is removed from the graph and any other neighbor node of s takes on the role of node v_1 . Note that it is not necessary to explore the last edge incident to s

2. Background on IP-over-WDM and p -Cycle Survivable Networks

as no further cycle can be formed; instead, both that last edge and the node itself are removed from the graph. And the biconnected components of the remaining graph are identified and the cycle finding procedure is applied again (to the new biconnected graph), beginning on another randomly chosen root node. The removal of an inspected node and its incident edges certainly has an accelerating effect on next node explorations, but [Gro03b] does not evaluate the overall complexity of the DFS all-cycle finding algorithm. He just indicates that the DFS all-cycle finding heuristic is based on a similar algorithm by [Joh75] for directed graphs, which showed a best-case space complexity of $O(|E|)$ for a time complexity of $O((|V| + |E|)(|C| + 1))$. And additionally, it points out that the DFS cycle finder includes some enhancements which improve its efficiency on bidirectional graphs.

2.3.2.4 Solving p -Cycle ILP Problems and Computational Complexity

While treating optimization problems and algorithms, it is natural to consider their inherent computational complexity. [Gro03b] gives a practitioner’s thumbnail guide to problem classes and indicates that ILP models exhibit an *NP-hard* behavior in practice. NP-hard algorithms inevitably have to examine all possible solutions, and it is not even possible to test the validity of a postulated solution in a polynomial time. The complexity can be reduced for “uni-modular” ILP problems as they can be solved as linear programming (LP) models without losing integrality. This is a significant advantage because LP does not constrain decision variables to integer values; and using the simplex algorithm, most LP models are provable polynomial time in practice.

The literature proposes many modeling languages and software packages such as `Lingo`, `LPSolve` and `AMPL/CPLEX` to design and solve large and complex LP and ILP problem instances. But in the specific case of p -cycle ILPs, those mathematical programming tools may become exhausting for large scale or highly connected networks having a huge number of candidate cycles. In those cases, p -cycle ILPs or corresponding combinatorial optimization problems require heuristic or meta-heuristic solutions. Some popular meta-heuristics are simulated annealing (SA), ant colony (AC), genetic algorithm (GA) or tabu search (TS). They all provide sub-optimal solutions but even though proof of optimality is not required, high standard quality solutions remain a goal.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

Another way to counteract the candidate pre-enumeration is to directly integrate cycle structure formation into the ILP problem itself. The principle follows the transportation-flow problem structure, which is a mathematical model that minimizes capacity cost requirements under the following constraints: any traffic generated at a sink for a given destination reaches the target node with no losses en route; no traffic flow *transits* either the origin or the destination, and no traffic terminates at the source or starts at the destination; and any traffic coming in a transshipment node on its way to the destination also goes out. From the cycle formation perspective, the last constraint known in the literature as the *flow conservation principle* imposes that each node has either 0 or 2 incident spans on each given cyclical structure. This specifically applies to p -cycle designs through enumeration-free ILP and column generation (CG) that will be discussed in chapter 5.

2.3.3 Configuration in IP-over-WDM Networks

Due to the mesh-like capacity efficiency of p -cycles, this research essentially considers them as a logical layer mesh-type survivability scheme. But to introduce such fundamentals as network elements and protection switching protocols, the following highlights their applicability to every level of the transport networking infrastructure.

2.3.3.1 Transmission System Layer and Generic Nodal Device

The realization of p -cycles in the DWDM transmission system layer assumes a protection at either the whole-fiber or the waveband granularity levels. Conventional rings can be converted into modular p -cycle structures by simply adding a straddling span interface unit on OADMs to access the protection channel on a ring. Accordingly, [Gro03b] proposed the generic ADM-like nodal device for p -cycle-based networking in Figure 2.24.

2.3.3.2 Configuration in the DWDM Logical Layer

DWDM systems allow one spare capacity signal unit to be manipulated so that the routing of working flows and the configuration of protection structures are not locked together as in conventional rings. In this context, OXCs can set-up and take down service paths as demand requires and separately configure and maintain a set of span-protecting p -cycles. Such p -cycles are established and managed at the logical channel, and can be fairly easily

2. Background on IP-over-WDM and p -Cycle Survivable Networks

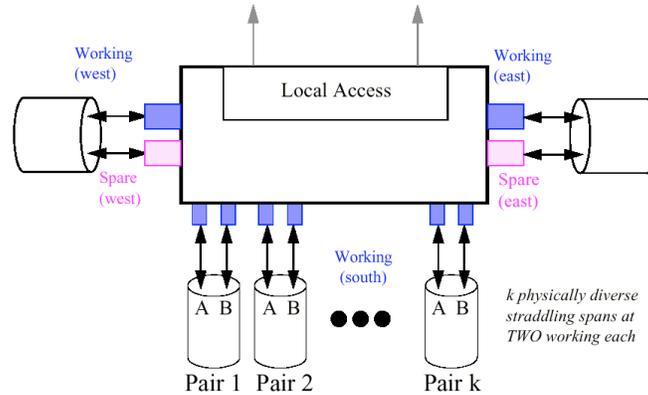


Figure 2.24: *ADM-like Nodal Device for p -Cycle-based Networking, adapted from [Gro03b, Gro02a].*

changed to adapt to shifting demand patterns or to multi-service priority schemes for access to protection cycles. But [SSG03a] emphasized that the logical layer implementation of p -cycles depends on whether wavelength conversion is possible at every OXC across working paths and along p -cycles, or working paths and p -cycles are (independently) transparent but do not use the same wavelength, or working paths and p -cycles are both transparent and the use same wavelengths. This differentiation corresponds to opaque, hybrid and transparent p -cycle types shown in Figure 2.25, and detailed in Section 4.2.

2.3.3.3 DWDM Protection Switching Protocols

[Sch05] indicates that APS-like protocols used for conventional rings can be fairly easily adapted to the context of span-protecting p -cycles. In the event of physical span or node failure, the nearest two neighbor nodes detect the failure and initiate APS-like protocol instances to recover affected paths. Those APS-like protocols are the p -cycle counter-parts of APS-like protocol instances used for conventional rings. More specifically, the APS protocol instance for 4-fibers BLSR in SONET is invoked by system layer p -cycles that operate at the fiber granularity level; and the APS protocol instance for 2-fibers BLSR in SONET is used by system layer p -cycles that operate at the waveband level of granularity. In the logical cross-connect layer, failure detection is performed at the wavelength level. The related APS-like protocol instance is equivalent to the one used for DWDM line-

2. Background on IP-over-WDM and p -Cycle Survivable Networks

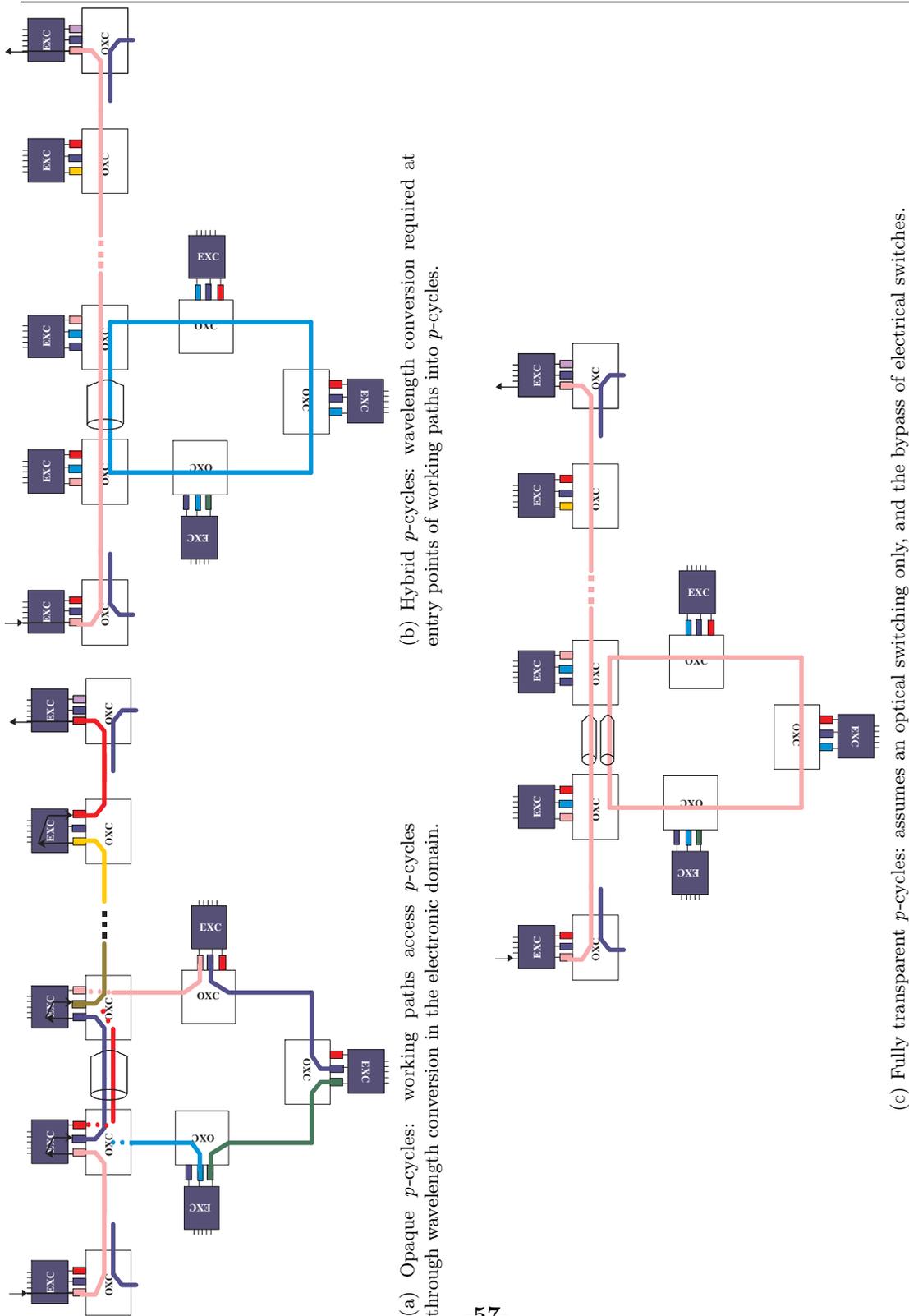


Figure 2.25: Specific Requirements for the Logical Layer Configuration of p -Cycles.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

switched rings, which are optical counter-parts of conventional SONET rings. (The reader may refer to the literature for specific details on pure APS and APS-like ring protocols that sustain APS-like p -cycle protocol analogies.)

2.3.3.4 Adaptation to the IP/MPLS Service Layer

In Section 2.2, we highlighted several reasons why survivability is preferable at either the system or the logical transport layers. Nevertheless, the following indicates the possibility for p -cycle implementation in the service layer that may be useful in several situations. More specifically, p -cycles have an inherent ability to recover from node failures that may be of some interest for the IP layer which is very prone to router failures due to their software related nature. p -Cycles may also help to avoid packet lost during routing table updates. And p -cycles planned under controlled oversubscription may help to prevent congestion and/or capacity impacts under pure IP-rerouting mode.

From practical implementation perspectives, plain IP is connectionless but [SG00a] brought attention to logical p -cycles that can be established as virtual circuits using MPLS or a small number of reserved IP addresses in routing tables. Such p -cycles are designed prior to failures in the network setup but consume zero capacity until used. If a packet's normal routing table entry indicates forwarding into a now-dead port, the packet is encapsulated with two fields: the p -cycle label on which the dead neighbor route belongs and the cost of the original pre-failure path for the IP packet. The encapsulated packet is routed along the p -cycle where each router continues the packet if it has no entry with a functional port for the true IP destination or if the cost of local continuing route option is greater than the cost in the encapsulated packet; otherwise, the " p -cycle packet" is decapsulated and the IP packet is normally routed from the current node. Figure 2.26 summarizes the operation principle above; Nortel implemented span restoration via IP-layer p -cycles and experienced about 10 ms restoration times and no packet loss [Gro03b].

2.4 p -Cycle Literature Survey

The p -cycle survivability technique was introduced in 1998 by Professor Grover and his former student Stamatelakis. Within the first years of existence, most publications were aimed

2. Background on IP-over-WDM and p -Cycle Survivable Networks

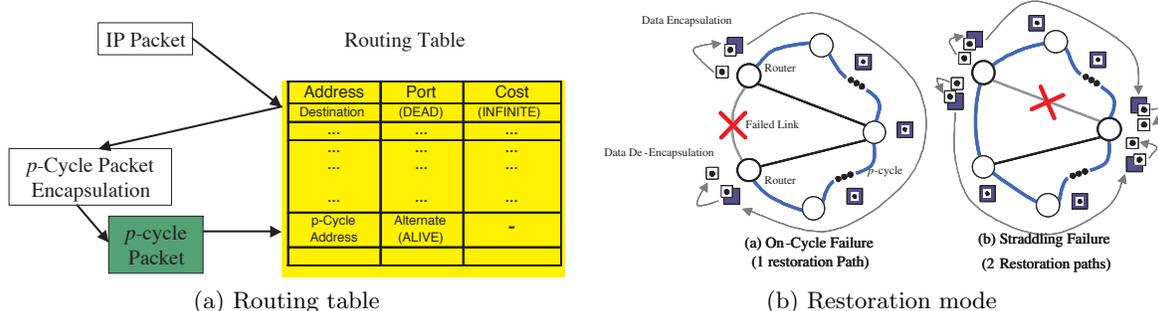


Figure 2.26: Operation of IP-Layer p -Cycles, adapted from [Gro03b, Gro02a].

at increasing the awareness and understanding of p -cycles in the community. More specifically, [GS98b] indicates that closed-paths can be self-organized out of the spare capacity in broadband mesh transport to form p -cycles and [GS98a, GS00, SG00b, KGSM03, BSGN03] explain the p -cycle behaviour in the event of span failures, and how this rides network demand growth and combines ring-like speed with mesh-like efficiency. Since the advent of p -cycles, related work moved from the fundamentals above to more advanced concerns.

2.4.1 Implementation and Operation in IP-over-WDM Networks

[Gro03b] indicates that p -cycles certainly could be centrally computed and configured based on mathematical methods. But another interesting option is to consider if the network can adaptively and continually self-organize, with a near-optimal set of p -cycles for whatever demand pattern and capacity configuration it currently finds. [Gro03b] proposes a distributed cycle pre-configuration protocol (DCPC), which is adapted and extended from SNH distributed mesh restoration algorithm, for the self-organization of p -cycles.

From the layering and partitioning standpoints, [SG99, SG00a] study the applicability of p -cycles to the IP layer. [SSG03a, SSG03b, Sch05] configure p -cycles in WDM networks with partial or full wavelength conversion capabilities at the network nodes, and adapt APS-like protection switching protocols to the context of p -cycles. [Gro02b, Gro03a] discuss new options and possibilities for ring-mesh hybrid architectures; and [SG04a] addresses the specific case of homogeneous optical networks bearing exactly two fibers per span exactly.

2.4.2 NEPC, Flow and FIPP Extended Protection Principles

[SG99, SG00a] describe initial studies on the rapid restoration of the IP layer based on virtual p -cycles and also highlight an ability to recover straddling flows transiting through a failed router. The important property is that each router may have a node-encircling p -cycle (NEPC) dedicated to its failure. This is a p -cycle that includes all logically adjacent nodes but not the protected node itself, so that the encircling structure intercepts all transiting flows through the subject node. Later studies by [DGG05, DGG07] revisit combined node and span protection strategies for multi-layer design with NEPCs, and will question the adaptability of physical layer p -cycles for router-level node failure protection.

[SG03b, SG03c, GK05, KGD05, KG05a] propose other extensions to the concept of p -cycles that may lead to practical methods for span and node failure recovery. The path-segment protection approach in [SG03b, SG03c] is based on the observation that every failed path intersects with one or more p -cycles anywhere along the path in question but both upstream and downstream of the respective failure. Such path-segment (or “flow-”) protecting p -cycles appear much more efficient than any other p -cycle type. Nevertheless, researchers almost abandoned the topic because of operational complexity incurred.

In contrast to path-segment protecting p -cycles, [GK05, KGD05, KG05a, JRBG07, BGK07] adapt p -cycles to end-to-end failure-independent path protection switching against either span or node failures. Failure-independent path-protecting (FIPP) p -cycles offer one specific pre-defined backup to each working path, with optical-path engineering prior to the failure, and switching requirements at the end-nodes only. Most prior work on regular span-protecting p -cycles was also conducted for FIPP p -cycles; the reader may want to refer to [Bal09] which is a thesis dedicated to FIPP advances.

2.4.3 Dual Failures, Optical Reach Control and Demand Uncertainty

The literature also studies the p -cycles’ ability to recover from dual span failures. More specifically, [CG05] tackles the dual span failure problem from the availability analysis perspective. As opposed to availability measures, [SCG04] discusses strategies for enhanced dual failure restorability with static or reconfigurable p -cycle networks. [KG05b] extends the study to multiple quality-of-protection (multi-QoP) policies including dual failure survivability service. Recently, [GG09] investigated a novel dedicated protection architecture

2. Background on IP-over-WDM and p -Cycle Survivable Networks

referred to as UPSR-like p -cycles, which could be used to enable high levels of dual failure restorability.

[SGA02, KSG04, KSG05] observe that loopback and break-in restoration through the entire circumference of p -cycle structures significantly contribute to the length of backup paths. For transparency and translucency purposes, [SGA02, KSG04] constrain candidate cycles under desired hop or circumference limits in the design. And [KSG05] directly limits restoration segment lengths, thereby searching p -cycles within all candidate structures while controlling optical reach in the restored network state.

[SG03a, LG04, SG04b, SG05, Gro05] face demand forecast uncertainty. [SG03a] exploits the forcer structure property of conventional rings to serve uncertain demands while minimizing redundancy of p -cycle networks. [SG04b, SG05] demonstrate that a p -cycle solution based on static forecasts also stands for the dynamic provisioning of survivable services through the concept of protected working capacity envelope (PWCE). And [Gro05] explains how the PWCE concept could be extended to address the needs of dynamic transparent optical networks with path-protecting p -cycles.

In other research, [GG07a] investigates whether it is feasible to design p -cycle networks such that the set of p -cycles used for protection could also serve as monitoring cycles (i.e. m -cycles) for rapid self-fault isolation. And [GG07c] compares p -cycles and p -trees in a unified mathematical framework. Although there has been an increased interest in p -cycles over the last decade, a significant amount of p -cycle related research is only aimed at developing mathematical modeling methods to solve p -cycle design problems, especially the FIPP p -cycle variant that involves much more complexity in network planning.

2.4.4 Questions Revisited and Research Methodology

This dissertation revisits transparent reach control in the restored network state and node protecting p -cycle questions; it explores the possibility for p -cycle protection at the glass-switched granularity level; and it addresses p -cycle design for large scale networks. For each of the topics, a comprehensive review of current practical methods that may lead to solutions is first conducted and new insights and principles are proposed to overcome the issues of prior related approaches. The new concepts are built into ILP design models (or

2. Background on IP-over-WDM and p -Cycle Survivable Networks

heuristics) and applied to several network instances. And the effectiveness and efficiency of results obtained in the experiments are compared to that of prior related methods.

2.4.4.1 Network Selection and Demand Patterns

Typically, thesis experiments will be performed on the same selection of real networks, given in Figures 2.28 to 2.30, for consistency. The **Havana** network in 2.27 consists of 17 nodes, 26 spans, 135 candidate cycles and two sets of demands: an original matrix of 58 demand relations distributed on the interval $[0..5]$, and another traffic matrix assuming connection requests between every single pair of nodes for a total of 136 demand relations with volumes distributed on the interval and $[10..100]$. The second test case network is the well-known **Cost239** pan-European network in Figure 2.28. It consists of 11 nodes, 26 spans, 55 non-zero demand-pairs distributed on $[1..11]$ and 3531 candidate cycles. The other test case instances are given in Figures 2.29, 2.30 and 2.31: the **Italy** instance has 13 nodes, 24 spans, 78 demand-relations distributed on $[0..10]$ and 557 candidate cycles; the **Bellcore** network has 15 nodes, 28 spans, 104 demand-pairs distributed on $[1..20]$ and 976 candidate structures; and the **Euro** network with 32 nodes, 42 spans, 323 demand-pairs uniformly distributed on $[0..2]$ and 699 candidate cycles.

If not mentioned, the demand patterns in Figures 2.28(b), 2.29(b), 2.27(b)-(c), 2.31(b) and 2.30(b) will pertain to the transport-level, as opposed to service-level. As well, it was earlier stated that demand working routes are decided prior to the design; a shortest distance weighted routing is typically applied under normal network states, resulting in total working channel-kms of 23,934 for **Havana** when considering the original set of demands; 166 channels for **Havana** network with the special case of hop-count routing applied to the original set of demands; 2,595,800 channel-kms for **Havana** network considering the larger set of demands; 137,170 channel-kms for **Cost239**; 62,232 channel-kms for **Italy**; 18,335 channel-kms for **Bellcore** and 246,375 channel-kms for **Euro**.

During the investigation of the large scale p -cycle problem, three other networks will be added to the selection above, i.e. **Havana**, **Cost239**, **Italy**, **Bellcore** and **Euro** accordingly referred to as *small and medium size networks*. Large scale test cases comprise the **Euronet** in Figure 2.32, with 19 nodes, 40 spans, 84,963 candidate cycles and 171 demand-pairs uniformly distributed on the interval $[0..10]$; the **Cselt** network in Figure 2.33, with 30 nodes,

2. Background on IP-over-WDM and p -Cycle Survivable Networks

56 spans, 387,740 cycle structures and 435 demand-pairs generated on a basis of node-pairs, following a uniform distribution on the interval [0..20]; and the 200-node challenge case in Figure 2.34, with 394 spans and 19,900 demand-pairs uniformly distributed on [0..10], randomly assembled from variations on the German graph of 50 nodes and 88 spans in `SNDLib` [SND06]. As a reference of normal network states, a single least hop working path routing is typically applied, resulting in 1968, 4159 and 693,731 working channels to be protected in `Euronet`, `Cse1t` and the 200-node network. Significantly, these three networks will be used to differentiate between large scale instances involving a huge number of candidates but manageable by the ILP solver, not importable into the ILP solver, or infinite and thus unknown. The 200-node challenge network is a cosmologically instance with as many as 15,307,626 cyclical structures under 18 hops alone.

2.4.4.2 Experimental Conditions and Preliminary Results

Each of the experiments consists of two steps: data preparation and ILP optimization. Preprocessing for routing, candidate cycle pre-enumeration and all other input parameters required by p -cycle ILPs are done on an Intel duo core processor running Mac OS X 10.5.8 at 2.8 GHz with 4 GB of 1067 MHz DDR3, using an adapted derivative of the TR Labs proprietary software [Dou01]. The software in question follows the prior-described DFS algorithm for cycle enumeration and route-finding; and it records on a disk-file the information about nodes, spans, traffic components, shortest distance route candidates for use under normal network conditions, and eligible cycle structures.

All of the ILP mathematical models are implemented in `AMPL 10.100` and solved using `CPLEX 10.1.0` on one of the following three computers: a 4-processor Sun UltraSparc III running at 900 MHz with 16 GB of RAM, one Pentium 3 GHz with 512 MB of RAM running Windows 2000, and an Intel duo core processor running Mac OS X 10.5.8 at 2.8 GHz with 4 GB of 1067 MHz DDR3. However, for consistency when discussing comparative results, we will assume one virtual machine which is a four-processor Sun UltraSparc III running at 3 GHz with 16 GB of RAM. All the running times will be scaled accordingly and the mixed integer programming gap of optimality (MIPGAP) being employed will be specified for each of the experiments.

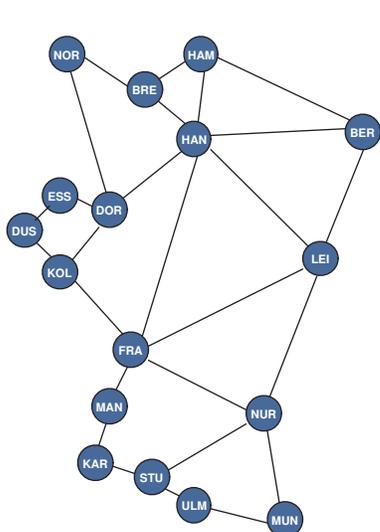
2. Background on IP-over-WDM and p -Cycle Survivable Networks

For reference, the second set of columns in Table 2.1 records the number of nodes, the number of spans and the demand bundles between node pairs for all test case networks. The third set of columns in Table 2.1 indicates candidate set cardinalities and DFS all-cycle finding completion times. And the fourth set of columns gives the number of variables and constraints involved in each ILP₁ instance, characterizes p -cycle minimum spare capacity design solutions (obtained with a MIPGAP of 10^{-4}), and indicates completion times.

Table 2.1: *Characteristics of Test Case Network Instances from the ILP₁ Perspective*

Problem Classes	Network Instances			DFS Cycle Finder		Conv p -Cycle Design, ILP ₁		
	N	S	D	P	run time	cmplx	solution	time
1	Havana							
	17	26	58	135	< 1sec	3,776	20,264 chan-km; 5 struct; 16 p -cycles.	< 1sec
	Cost239							
	11	26	55	3531	< 1sec	95,468	85,640 chan-km; 9 struct; 19 p -cycles.	~ 20 sec
	Italy							
	13	24	78	557	< 1sec	14,046	55,654 chan-km; 8 struct; 98 p -cycles.	< 1sec
1	Bellcore							
	15	28	104	976	< 1sec	28,445	12,629 chan-km; 13 struct; 192 p -cycles	< 1sec
	Euro							
	32	42	323	699	< 1sec	30,268	235,206 chan-km; 17 struct; 99 p -cycles	< 1sec
	Euronet							
	19	40	171	84,963	~ 2.7 sec	1,699,356	834 channels; 19 struct; 52 p -cycles.	4 min
2	Cselt							
	30	56	435	387,740	~ 30.8 sec	22,101,461	var and constr exceed the 16GB memory limits	
3	200-node							
	200	394	19,900	The space of all distinct simple cycles multiply towards infinity (under 18 hops)			15,307,626	6,046,513 $\times 10^3$

2. Background on IP-over-WDM and p -Cycle Survivable Networks



(a) Topology: 17 nodes, 26 spans, 135 candidate cycles

NOR	BRE	HAM	HAN	BER	ESS	DOR	DUS	KOL	LEI	FRA	MAN	KAR	STU	ULM	MUN	NUR	
0	2	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	NOR
.	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	BRE
.	.	3	1	0	1	1	1	1	2	1	0	0	1	0	1	0	HAM
.	.	.	2	0	4	1	4	3	5	0	0	1	0	0	2		HAN
.	.	.	.	0	0	0	0	2	1	0	0	0	0	0	1	1	BER
.	1	1	1	0	0	0	0	0	0	0	0	0	ESS
.	1	4	3	3	0	0	0	0	0	0	0	DOR
.	2	0	1	0	0	1	0	0	0	0	DUS
.	3	5	0	0	1	0	0	0	0	KOL
.	3	0	0	1	1	1	1	1	LEI
.	1	1	2	2	2	1		FRA
.	1	1	0	0	0	0	MAN
.	1	0	1	0		KAR
.	1	1	1		STU
.	1	0		ULM
.	1		MUN
.		NUR

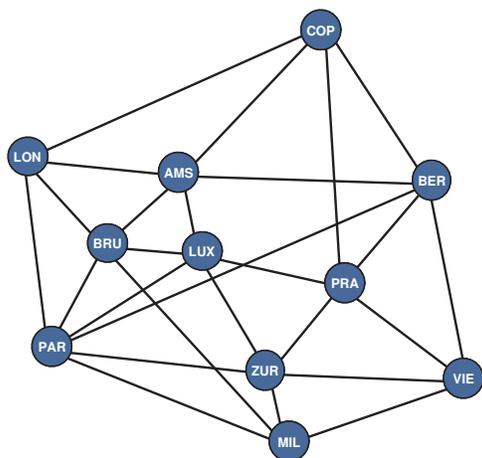
(b) Original Traffic Matrix, 58 demand-pairs distributed on [0..5]

NOR	BRE	HAM	HAN	BER	ESS	DOR	DUS	KOL	LEI	FRA	MAN	KAR	STU	ULM	MUN	NUR	
	11	40	44	62	40	68	59	37	61	82	43	40	28	60	45	45	NOR
.		74	60	16	16	74	34	64	79	85	33	77	25	98	42	60	BRE
.	.		11	17	27	65	38	70	96	73	27	80	97	88	38	36	HAM
.	.	.		53	59	46	91	65	44	74	75	37	52	83	32	88	HAN
.	.	.	.		23	12	100	48	52	27	30	80	12	80	34	46	BER
.		69	22	72	19	70	94	50	99	41	78	99	ESS
.		95	63	26	14	44	92	46	34	69	73	DOR
.		31	100	83	87	79	72	26	79	28	DUS
.		78	14	70	49	19	94	71	59	KOL
.		79	17	54	62	44	34	57	LEI
.		29	22	89	68	66	35	FRA
.		11	76	12	28	18	MAN
.		57	88	77	50	KAR
.		83	47	23	STU
.		20	52	ULM
.		49	MUN
.		NUR

(c) Large Traffic Matrix, 104 demand-pairs distributed on [11..100]

Figure 2.27: Havana Test Case Instances.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

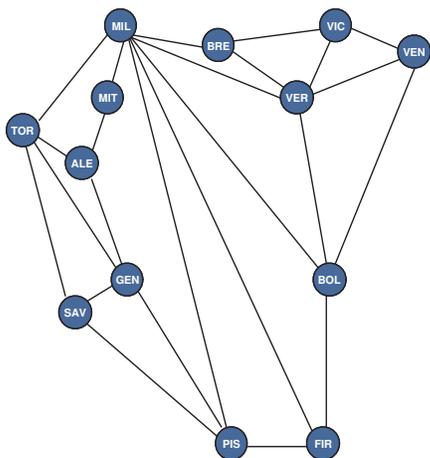


(a) Topology: 11 nodes, 26 spans, 3531 cycles

	PAR	MIL	ZUR	PRA	VIE	BER	AMS	LUX	BRU	LON	COP
PAR	0	5	6	1	2	11	5	1	7	10	1
MIL	.	0	6	1	3	9	2	1	2	3	1
ZUR	.	.	0	1	3	11	3	1	6	3	1
PRA	.	.	.	0	1	2	1	1	1	1	1
VIE	0	9	1	1	1	2	1
BER	0	8	2	6	8	3
AMS	0	1	4	5	1
LUX	0	1	1	1
BRU	0	4	1
LON	0	1
COP	0

(b) Traffic Matrix, 55 demands distributed on [1..11]

Figure 2.28: *Cost239 Test Case Network.*



(a) Topology: 13 nodes, 24 spans, 557 cycles

	TOR	ALE	SAV	GEN	MIL	MIT	BRE	VER	VIC	VEN	BOL	FIR	PIS
TOR	.	4	7	8	3	3	7	8	7	10	10	4	3
ALE	.	.	9	10	8	2	2	9	1	5	0	9	5
SAV	.	.	.	8	9	8	4	1	3	0	2	6	6
GEN	0	1	10	8	5	3	7	1	6
MIL	4	9	2	4	9	6	3	7
MIT	7	6	9	9	5	2	7
BRE	5	6	7	4	4	8
VER	5	9	5	4	0
VIC	9	5	5	5
VEN	8	6	3
BOL	2	7
FIR	0
PIS

(b) Traffic Matrix, 78 demand-pairs distributed on [0..10]

Figure 2.29: *Italy Test Case Network.*

2. Background on IP-over-WDM and p -Cycle Survivable Networks

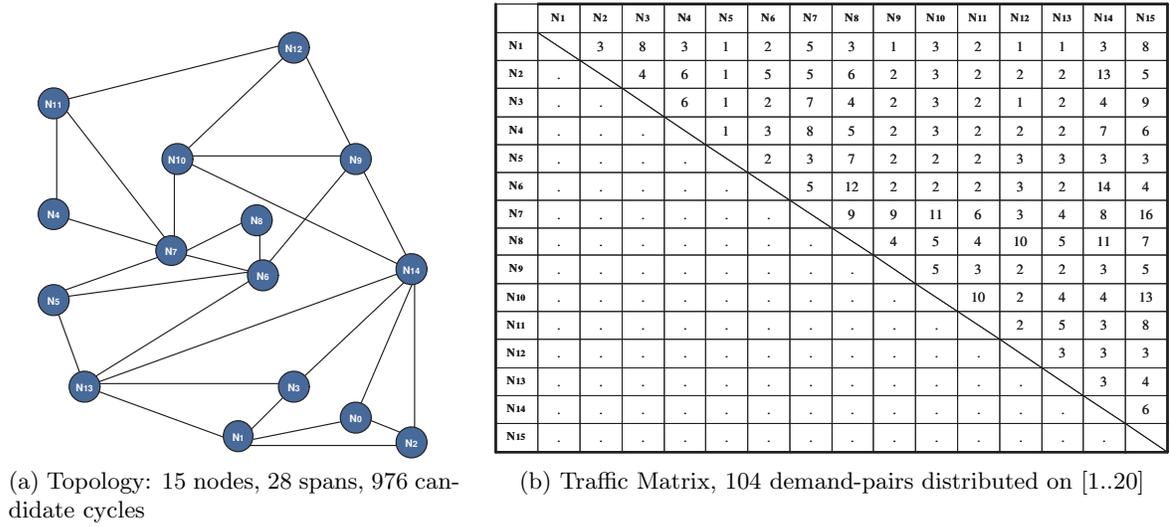
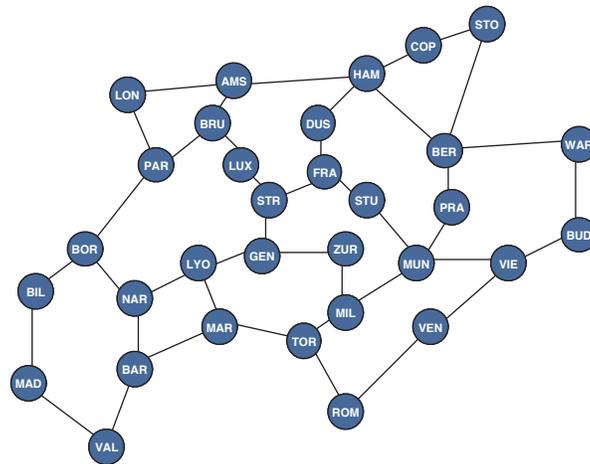


Figure 2.30: Bellcore Test Case Network.



(a) Topology: 32 nodes, 42 spans, 699 candidate cycles

2. Background on IP-over-WDM and p -Cycle Survivable Networks

	AMS	BAR	BER	BIL	BOR	BRU	BUD	COP	DES	FRA	GEN	HAM	LON	LUX	LYO	MAD	MAR	MIL	MUN	NAR	PAR	PRA	ROM	STO	STR	STU	TOR	VAL	VEN	VIE	WAR	ZUR		
AMS	0	2	1	0	2	0	1	1	2	2	1	1	0	1	0	0	2	2	2	2	1	2	1	1	1	1	0	1	1	0	1			
BAR		0	1	1	0	0	1	0	0	1	2	2	2	1	1	2	2	0	1	2	1	2	2	0	1	2	0	0	2	2	1	0		
BER			0	2	0	1	2	1	1	2	1	2	1	1	2	2	1	0	1	2	2	0	1	1	2	2	2	1	0	2	1	2		
BIL				0	2	0	0	1	2	2	2	2	2	1	1	1	1	1	0	0	2	1	2	0	0	0	0	0	2	1	0	1		
BOR					0	1	1	0	0	1	2	2	2	2	2	2	0	1	0	0	0	1	1	0	0	2	0	0	1	0	1	0		
BRU						0	2	1	1	0	0	2	2	2	1	0	0	2	0	1	1	0	1	1	2	1	2	1	1	2	2	2		
BUD								0	1	2	0	1	2	1	1	1	1	2	0	0	0	2	1	0	2	1	0	2	2	2	1	0		
COP									0	0	0	0	1	1	1	0	1	2	0	0	0	0	1	0	2	2	2	1	2	2	2	1	0	
DES										0	2	0	2	0	1	1	0	2	0	1	1	2	0	2	2	1	1	1	1	0	0	2	0	
FRA											1	2	2	2	0	0	1	0	0	0	2	0	1	1	2	2	0	2	0	0	0	2		
GEN												2	1	2	0	1	1	0	1	2	1	2	0	1	2	0	0	0	1	2	1	0		
HAM													0	0	1	0	2	1	2	1	2	1	0	2	0	1	2	0	2	0	1	0		
LON														0	0	2	0	2	2	1	1	2	0	0	2	0	1	0	2	1	2	1		
LUX															1	0	1	1	1	1	2	1	2	2	2	2	1	2	2	0	1	1		
LYO																0	2	1	1	1	1	2	0	0	2	1	2	2	2	2	0	0		
MAD																	0	2	1	0	1	2	2	1	0	0	2	2	2	1	0	2		
MAR																		2	1	2	2	0	2	2	2	2	0	1	2	0	2	1		
NEL																			0	0	0	1	1	0	2	2	0	2	0	0	2	0		
MUN																				2	1	0	1	0	1	0	2	1	2	0	0	2		
NAR																					2	1	2	1	0	2	2	1	0	0	2	2		
PAR																						2	2	2	1	1	1	2	1	2	2	0		
PRA																							2	2	2	0	2	2	0	0	1	0		
ROM																								0	1	0	2	1	0	1	2	2		
STO																																		
STR																																		
STU																																		
TOR																																		
VAL																																		
VEN																																		
VIE																																		
WAR																																		
ZUR																																		

(b) Traffic Matrix, 323 demand-pairs uniformly distributed on [0..2]

Figure 2.31: Euro Test Case Network.

2. Background on IP-over-WDM and p -Cycle Survivable Networks

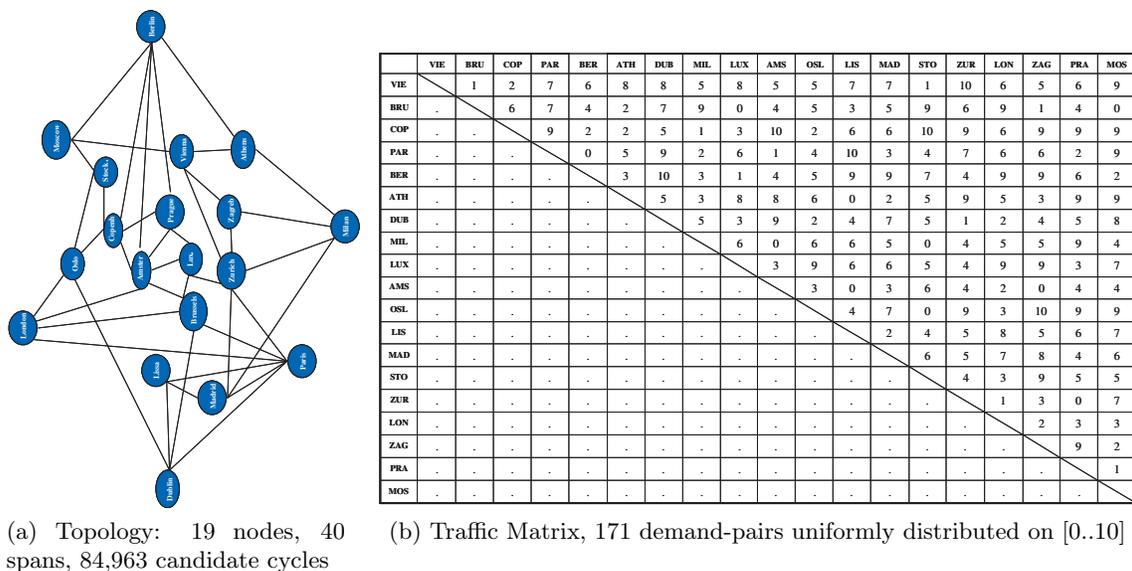


Figure 2.32: Euronet Test Case Network.

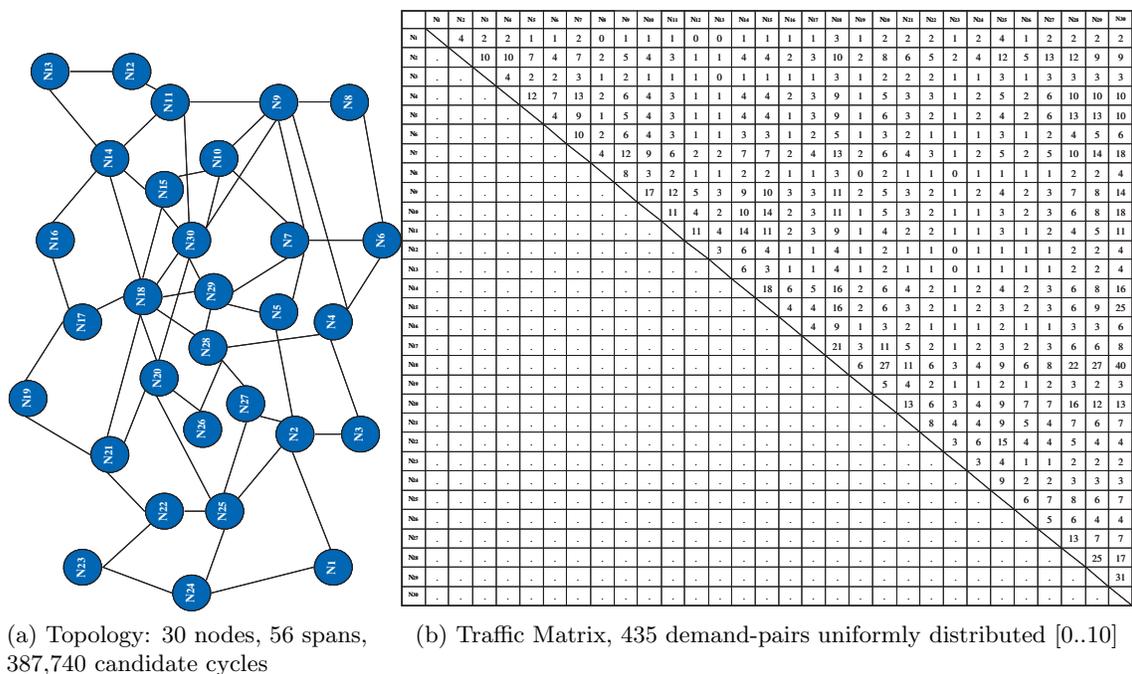


Figure 2.33: Cselst Test Case Network.

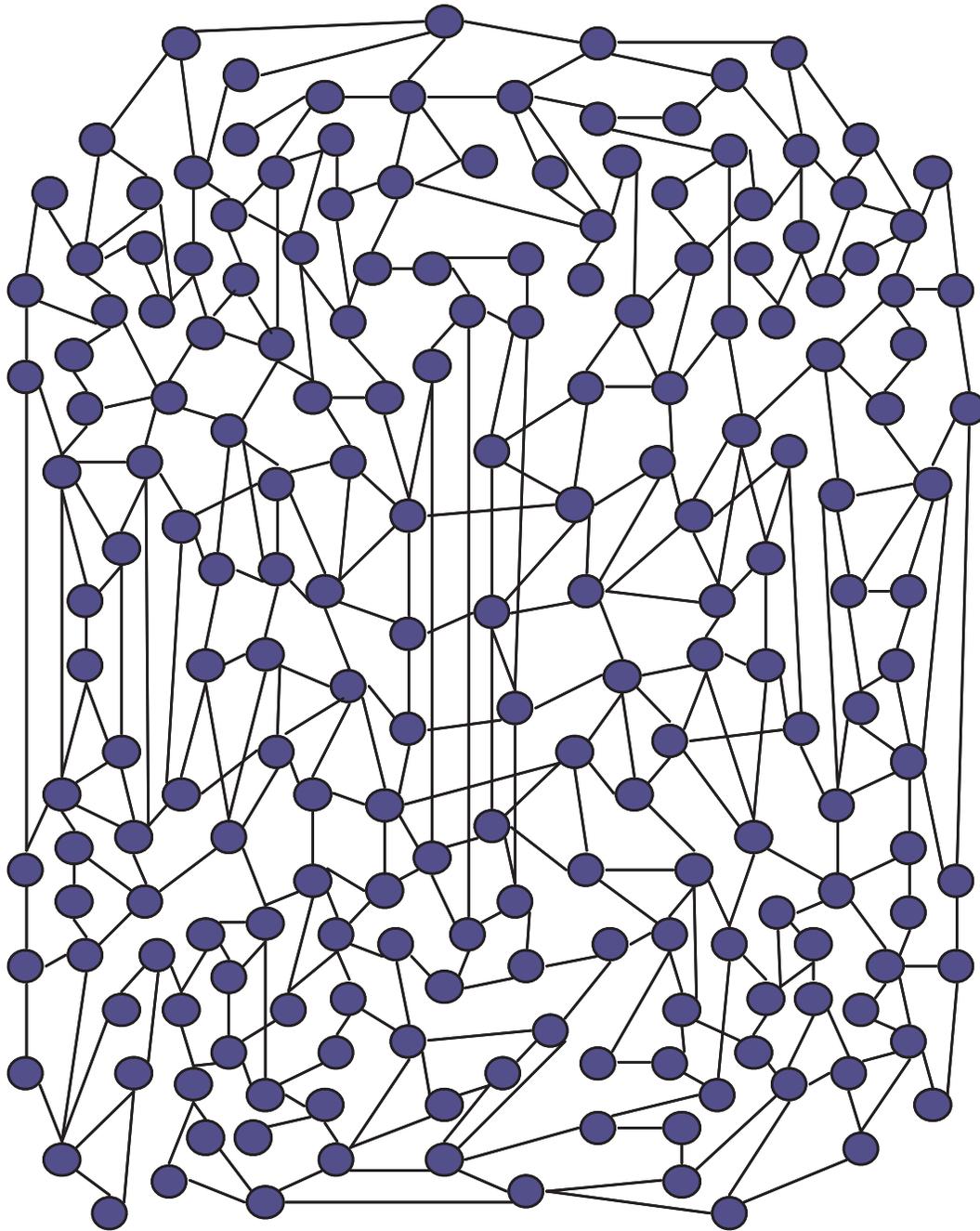


Figure 2.34: *The 200-node Challenge Topology: 200 nodes, 394 spans; working capacities atop spans.*

Chapter 3

Node-Protecting p -Cycles

This chapter is adapted and extended from [OG08b, GO09, OG10b, OG10c, OBG09], all of which pertain to the topic of node failure recovery using p -cycles.

- More specifically, the chapter shows that viewed in a generalized two-hop framework for node failure recovery, ordinary p -cycles actually have a very *high* inherent ability to restore paths transiting through a failed node. With very little extra spare capacity, if any, the principle is also amenable to explicit design of networks for 100% node and span failure protection with a single efficient set of p -cycles which support both functions. This is very different than the often-prevailing assumption that “ordinary” span-protecting p -cycles offer no node protection, or only the same protection as a BLSR ring embodies.
- In a further contribution, the present chapter also reveals that p -cycle intrinsic node protecting capabilities are extendable to a novel concept of *two-hop protecting p -cycles*, no longer distinguishing between node and span failures to achieve a complete network restorability. Indeed, two-hop paradigms for recovery of affected paths transiting through failed nodes and two-hop protecting p -cycles provide an attractive option for future network operators in that ordinary span-protecting p -cycles are more localized, fast acting, and simple to plan than any other option.
- Another research aims to simplify the design of failure-independent path-protecting (FIPP) p -cycles. Related study first relaxes the constraint of failure-independency that greatly complicates FIPP-based designs in practice. And we discuss how such a generalization approximates (or can be fixed to provide) FIPP solutions.

The following is an outline to the chapter. Section 3.1 discusses the relevance of such a research study, and acknowledges various extensions to the original concept of p -cycles that

may lead to node failure protection. Section 3.2 unifies all of the prior related principles through a general criterion, inherent to “ordinary” p -cycles but missed to-date in the literature. Section 3.3 extends the generalized approach to a novel *two-hop segment protection* paradigm, no longer requiring to distinguish node from span failures in the design. Section 3.4 proposes *general path-protecting p-cycles* as an alternate strategy for easily obtaining FIPP p -cycle solutions. And Section 3.5 summarizes chapter contributions and outline originality.

3.1 The Question of Node-Protecting p -Cycles

Network survivability design is primarily focused on recovery from span failures because the frequencies of fiber cable cut events are hundreds to thousands of times higher than corresponding reports of transport layer node failures. If OXCs tend to be highly robust and well protected, IP/MPLS routers still suffer downtimes about as frequently as span failures due to software patches, upgrades or even crashes. Such outages are particularly harmful because each specific node failure involves the simultaneous failure of all node-incident edges. Thus, the question of node failure protection remains of concern to the domain of network survivability.

End-to-end path-protecting survivability techniques intrinsically respond to *intermediate* node failures arising somewhere along the working paths. And corresponding levels of node failure restorability (i.e., R_{1-node}) depend on backup channel-capacities and node-disjointness considerations in the shared risk link groups (SRLGs). In contrast, it is much more challenging to provide node recovery functionalities using span-protecting architectures because these are based on the deployment of a set of backup path-segments between the end-nodes of the span failures in question.

The original intention with *span protecting* or “ordinary” p -cycles was efficient and fast protection against single span failures. A common misunderstanding is that ordinary span-protecting p -cycles offer no form of node protection. More correctly, p -cycles inherently offer the same protection to *on-cycle* paths traversing a failed node, as does a BLSR ring with respect to paths on the cycle [SG00a, Sch05]. What has remained less clear is how to protect paths, transiting through a node on a p -cycle, which have *straddling relationship*

to the respective p -cycle. To protect straddling paths against node failures as well, there has been various extensions to the basic node-protecting property of p -cycles.

3.1.1 Node-Encircling p -Cycles (NEPCs)

One idea for node protection using p -cycles is the *node-encircling p -cycle* (NEPC) principle studied and developed in [SG00a, DGG05, DGG07]. As illustrated in Figure 3.1, a p -cycle is said to be an NEPC for a given “encircled” node if it contains all neighbor-nodes of the encircled node, but not the given node itself. The key property is that an NEPC intercepts any flow transiting the encircled node and hence, with suitable spare capacity, can reroute all affected transiting flows when the encircled node fails.



(a) The “blue” p -cycle is an NEPC for node G (b) The NEPC intercepts any flow transiting through G

Figure 3.1: Illustration of the NEPC Concept, adapted from [Gro03b, Dou06].

But because an NEPC does not include the protected node itself, this approach does not exploit the inherent reaction p -cycles can have against on-cycle node failures. Fairly often as well, some nodes may have no (simple) NEPC in a graph sense, especially in sparser networks. *Non-simple* candidate cycles crossing a span or a node more than once can be considered, but this adds greatly to the operational and conceptual complexity. Also, designing a separate set of NEPCs generally requires significantly more spare capacity in the complete design because NEPCs provide for node recovery separately from other (span-protecting) p -cycles.

3.1.2 Path-Segment Protecting p -Cycles

Another line of work partly motivated by including node protection has led to extensions of the whole p -cycle concept into *path-segment* (or simply, “flow-”) *protecting p -cycles*

[SG03b, SG03c]. As shown in Figure 3.2, the principle is to observe that every p -cycle will also happen to intersect a number of working flows upstream and downstream. If the respective working path-segments are viewed as *virtual spans*, any intermediate node or span failure along an intersecting path-segment will be viewed as causing the failure of a virtual span. So conventional p -cycle switching operations can be applied.

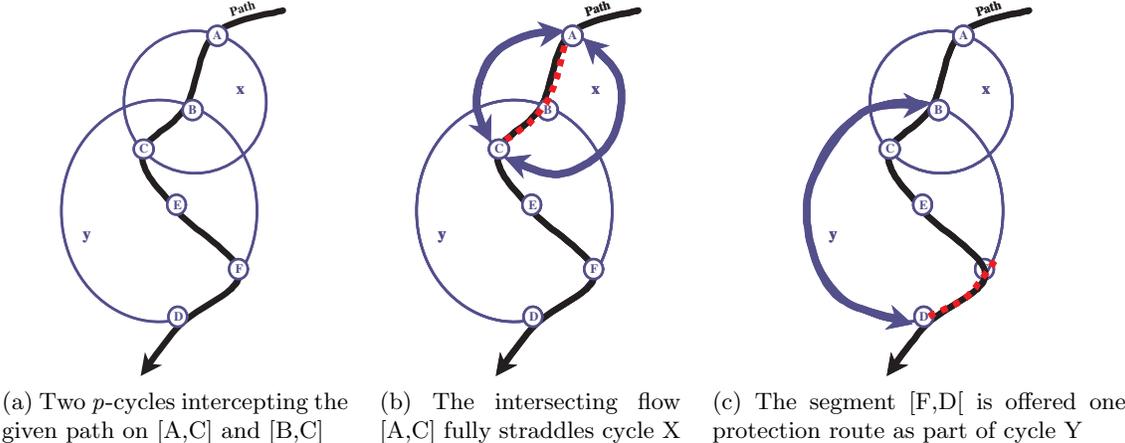


Figure 3.2: Flow-Protecting p -Cycles X and Y Handling Path-Segments $]A,C[$ and $]B,D[$.

The preliminary study in [OG08b] shows that very high levels of node failure restorability are achievable by applying the path-segment view to ordinary span-protecting p -cycles. But the method of flow-protecting p -cycles require advanced inter-nodal signaling or centralized management to activate right restoration actions, which can be different depending on where the failed path-segment is disrupted. For example, Figure 3.2 shows two p -cycles X and Y respectively intercepting segments $]A,C[$ and $]B,D[$ of a working path. In 3.2(c), Y handles the flow-segment $]B,D[$ but with different protection relationships along $]B,F[$ vs. $]F,D[$. Because two restoration routes are available for $]B,F[$ and only one for $]F,D[$, it requires a true knowledge of where the failure occurred to take proper restoration actions.

3.1.3 Failure-Independent Path-Protecting (FIPP) p -Cycles

FIPP p -cycles operate like conventional p -cycles but they are chosen so that each protects a set of end-to-end paths that are mutually span-failure disjoint between end-nodes on the FIPP structure [GK05, KG05a, KGD05, BGK07, OBG09]. The end-nodes of demand

relations are responsible for failure detection and the restoration actions required for failure recovery. The location of the actual failure is not important and only the knowledge that a failure has occurred is needed for the end-nodes to perform the switching action that directs affected working paths onto the protection paths provided by FIPP p -cycles. So with proper node-disjointness constraints, FIPP p -cycles stand as a valid alternative to node failure protection. But this implies switching from span- to path-oriented paradigms, and FIPP network planning in itself is not a trivial task, as will be shown in Section 3.4.

3.2 A Simple Generalized Approach to Node Failure Recovery with Span-Protecting p -Cycles

To date, researchers have overlooked the fact that the BLSR-like loopback reaction ordinary p -cycles make to restore on-cycle flows transiting through a failed node is also applicable to straddling flows failing at a p -cycle node, *if the two spans adjacent to the failure node both end on other nodes on the same p -cycle*. In hindsight, this is always true for on-cycle flows transiting a node and allows protection for additional cases in Figure 3.3; so this is a generalization of the BLSR-like node protection condition.

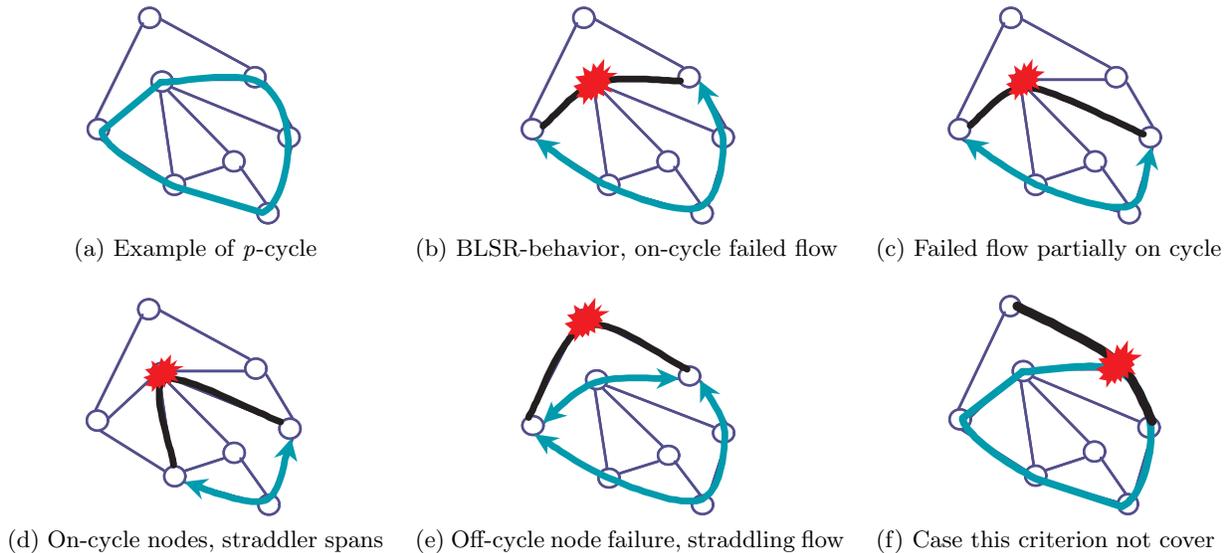


Figure 3.3: Intrinsic “Two-hop” Node-Protecting Capabilities of p -Cycles.

3.2.1 New Insights and Principle

Figure 3.3 illustrates how an ordinary span-protecting p -cycle can restore any two-hop path-segment intersecting the cycle structure upstream and downstream of a given failed node, whether or not the two-hop segment is entirely on the protecting structure. Figure 3.3(a) gives a p -cycle under the normal network state, and Figures 3.3(b)-(e) show how that p -cycle can be used to react in a previously overlooked way under node failure circumstances. *The only requirement is that the end-nodes of the two-hop segment are on the same p -cycle as each other.*

The failure scenarios consider whether the two-hop segment is entirely on the protecting structure as in 3.3(b), or if the p -cycle crosses only one of the spans of the two-hop segment as in 3.3(c), or if both spans of the two-hop segment straddle the cycle as in 3.3(d), or if only the end-nodes of the two-hop flow are part of the cycle structure as in 3.3(e). One way to think about this is to consider any two-hop flow as a kind of “virtual span”: the three cases in 3.3(b)-(d) are all equivalent to on-cycle (span) failures while the situation in 3.3(e), in which neither of the two spans comprising the two-hop segment nor the failed node are part of the cycle, corresponds to a p -cycle reacting to a straddling (span) failure. Figure 3.3(f) represents another class of situation where at least one end-node of the two hop-segment is not part of the protecting cycle; this cannot be covered by the novel criterion.

In practice, the two-hop standpoint retains the simplicity of operations of ordinary p -cycles and employs only one set of candidate structures in a complete design for both 100% span and node failure protection. Compared to related concepts, flow-protecting p -cycles generalize the two-hop strategy in the sense that path-segments may freely go from one or more spans to entire working paths. By restricting failed flows as if they were two-hop segments, the only (but not insignificant) requirement is the same simple and local type of failure detection and pre-defined switching plans as for span failures. On the other hand, the two-hop strategy selects a subset of ordinary cycles that covers all the neighborhood of a failed node. In one sense, this is a substitute for the node-encircling constraint; the merit is to offer a two-hop strategy much more flexible and efficient than NEPCs.

3.2.2 ILP Mathematical Formulations

Table 3.1 summarizes the ILP models developed within the chapter. ILP₂₋₆ pertain to two-hop flows, full flow-segments and NEPCs; they are aimed at maximizing R_{1-node} in p -cycle designs with controlled or no penalties over min-cost requirements, or at reversing the problem to achieve full R_{1-node} using the least possible total spare capacity.

Table 3.1: *ILP Mathematical Formulations for Node-Protecting p -Cycles*

Model	Problem Description	ILP Formulation
ILP ₂	Maximum R_{1-node} , given a 100% span restorable design as input.	Assume an existing set of p -cycles and working paths—equations (3.1)-(3.8).
ILP ₃	p -Cycle minimum spare capacity planning under R_{1-node} maximization.	Bi-criteria min capa and node unrestorability—equations (3.2)-(3.11).
ILP ₄	R_{1-node} maximization with controlled penalties over minimum capacity.	Merge ILP ₁ and ILP ₂ <i>plus</i> extra budget—equations (3.1)-(3.8) and (3.10)-(3.12).
ILP ₅	Full protection against both node and span failures.	Min spare capa for full R_{1-node} —equations (2.1)-(2.3) from ILP ₁ <i>plus</i> (3.2)-(3.4) and (3.13).
ILP ₆	Maximum possible R_{1-node} under minimum spare capacity with NEPCs.	Prevent the scarcity of NEPCs in ILP ₅ —equations (3.2)-(3.8), (3.10)-(3.11) and (3.14).
ILP ₇	Minimum spare capacity design using two-hop protecting p -cycles.	Equations (3.15)-(3.20).
ILP ₈	Minimum space capacity design for general path-protecting p -cycles (GPP).	Reference values and actual cycles for use in ILP ₉₋₁₀ ; equations (3.21)-(3.24).
ILP ₉	Imposing the failure independence constraint to an existing GPP design.	Assume an existing set of p -cycles and working paths—equations (3.25)-(3.28).
ILP ₁₀	Similar to ILP ₂ , but for GPP.	Equations (3.1), (3.4)-(3.8) and (3.29)-(3.31).
ILP ₁₁	Similar to ILP ₃ , but for GPP.	Eqns (3.4)-(3.9), (3.22)-(3.24) and (3.29)-(3.31).
ILP ₁₂	Similar to ILP ₄ , but for GPP.	Equations (3.1), (3.4)-(3.8), (3.12), (3.22)-(3.24) and (3.29)-(3.31).
ILP ₁₃	Similar to ILP ₅ , but for GPP.	Equations (3.4), (3.13), (3.16)-(3.19), (3.21)-(3.24) and (3.29)-(3.31).

3.2.2.1 ILP₂ Maximizing R_{1-node} in a Conventionally Designed p -Cycle Network

An assumption of ILP₂ is to keep as is the routing of working paths, the spare capacity and the p -cycles selected in an otherwise conventional p -cycle minimum spare capacity design.

Sets

- N is the set of nodes in the network, indexed by k .

- D is the set of demands, indexed by r . All units for a given demand relation r are assumed to be carried over the same working route under normal network states.
- P is the set of candidates, determined by a pre-processing method and indexed by p .

Input Parameters

- d^r is the number of units of capacity for demand relation r .
- $\vartheta_k^r \in \{0, 1\}$ encodes end-nodes for demand relation r . $\vartheta_k^r = 1$ if node k is either the origin or the destination of r , $\vartheta_k^r = 0$ otherwise.
- $\varepsilon_k^r \in \{0, 1\}$ indicates which nodes are on the working route of demand relation r ; $\varepsilon_k^r = 1$ if r crosses k en route, and $\varepsilon_k^r = 0$ otherwise.
- $\mu_k^{p,r} \in \{0, 1, 2\}$ indicates how many protection routes within the cycle p can handle the two-hop segment for demand relation r . If p intersects a two-hop segment upstream and downstream of k , $\mu_k^{p,r} = 2$ if k is off-cycle and $\mu_k^{p,r} = 1$ if k is on-cycle; otherwise $\mu_k^{p,r} = 0$.

Decision Variables

- η^p is the number of unit-sized copies of p -cycle p , in the design.
- $n_k^{p,r}$ is the number of copies of cycle p allocated to demand r to prevent node k failures.
- $\theta_k^{p,r}$ is the number of capacity units for demand-pair r effectively rerouted within p -cycle p when node k fails.
- Λ_k, Γ_k and Θ_k are for statistics on affected, transiting and recovered traffic, in the events of node k failure. The following inequality is always true: $\Lambda_k > \Gamma_k > \Theta_k$.

ILP₂ is given by equations (3.1) to (3.8). The objective function (3.1) maximizes node failure restorability. Equation (3.2) assigns protection segments to working paths transiting the intermediate node of two-hop segments, through p -cycles involved in the conventional design. Equation (3.3) ensures that only *intersecting flows* that are potentially restorable

are assigned protection paths. Equation (3.4) constrains the assignment of protection path-segments under the actual copies of p -cycles. Through equation (3.5), no credit is given to potentially protected paths that would exceed the actual demand volume present.

Equations (3.6)–(3.8) are for statistics only. They respectively compute the demand volume affected by a given node outage, the amount that was transiting the failed node and that is (thus) potentially restorable, and the number of working paths that are effectively protected in the design. Node failure restorability is given by $R_{1\text{-node}} = \sum_{k \in N} \Theta_k / \sum_{k \in N} \Gamma_k$.

$$\text{Maximize } \sum_{k \in N} \Theta_k. \quad (3.1)$$

$$\theta_k^{p,r} \leq n_k^{p,r} \cdot \mu_k^{p,r}, \quad \forall r \in D, \forall k \in N, \forall p \in P. \quad (3.2)$$

$$n_k^{p,r} \leq \mu_k^{p,r} \cdot \infty, \quad \forall r \in D, \forall k \in N, \forall p \in P. \quad (3.3)$$

$$\eta^p \geq \sum_{r \in D} n_k^{p,r}, \quad \forall k \in N, \forall p \in P : \varepsilon_k^r = 1. \quad (3.4)$$

$$\sum_{p \in P} \theta_k^{p,r} \leq d^r, \quad \forall r \in D, \forall k \in N : \varepsilon_k^r = 1 \text{ and } \vartheta_r^k = 0. \quad (3.5)$$

$$\Lambda_k = \sum_{r \in D} \varepsilon_k^r \cdot d^r, \quad \forall k \in N. \quad (3.6)$$

$$\Gamma_k = \sum_{r \in D: \vartheta_r^k = 0} \varepsilon_k^r \cdot d^r, \quad \forall k \in N. \quad (3.7)$$

$$\Theta_k = \sum_{r \in D, p \in P: \vartheta_r^k = 0} \varepsilon_k^r \cdot \theta_k^{p,r}, \quad \forall k \in N. \quad (3.8)$$

3.2.2.2 ILP₃ for Bi-criterion Optimization of Spare Capacity and $R_{1\text{-node}}$

Rather than maximizing node failure recovery in a pre-planned 100% span restorable p -cycle network, ILP₃ nudges the minimum spare capacity solution to support simultaneously the maximum feasible level of $R_{1\text{-node}}$. Setting a suitably small α , ILP₃ is achievable using the bi-criteria objective (3.9) associated with constraints (3.2)–(3.8) and (3.10)–(3.11). Equations (3.10)–(3.11), recalled for clarity purposes, are identical to (2.2)–(2.3) from ILP₁.

Additional symbols were essentially used in ILP₁ formulation where S was the set of spans in the network, indexed by i for failing spans and j for surviving spans or spans

in general; C_j was the km-cost of each channel on span j ; w_i was the number of working channels to be protected on span i ; and s_j was the number of spare channels on span j .

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j + \alpha \cdot \sum_{k \in N} (\Gamma_k - \Theta_k). \quad (3.9)$$

$$w_i \leq \sum_{p \in P} x_i^p \cdot \eta^p, \quad \forall i \in S. \quad (3.10)$$

$$s_j = \sum_{p \in P: x_j^p=1} \eta^p, \quad \forall j \in S. \quad (3.11)$$

3.2.2.3 ILP₄ Maximizing R_{1-node} with Controlled Capacity Penalties

One might want to assert that maximum R_{1-node} is subject to an allowable extra budget ξ which is relative to the minimum spare capacity cost B for 100% restorability against single span failures. Accordingly, ILP₄ is defined by equations (3.1)-(3.8) and (3.10)-(3.12).

$$\sum_{j \in S} C_j \cdot s_j \leq B \cdot (1 + \xi). \quad (3.12)$$

3.2.2.4 ILP₅ for Full Node Failure Protection

In a different way, one can ask what is the minimum spare capacity required to guarantee 100% restorability against both node and span failures. This is equivalent to ILP₅, which combines all equations from ILP₁ to constraints (3.2)-(3.4) and (3.13). Equation (3.13) specifically transforms inequality (3.5) to achieve the needs for full node failure recovery.

$$\sum_{p \in P} \theta_k^{p,r} = \varepsilon_k^r \cdot d^r, \quad \forall k \in N : \vartheta_r^k = 0. \quad (3.13)$$

3.2.2.5 Adaptations for BLSR-like Behavior, Flow p -Cycles and NEPCs

All prior-defined ILPs are applicable to the cases of BLSR-like behavior, NEPCs and flow-protecting p -cycles. The principle is to recognize that each p -cycle provides conventional span failure protection, but the p -cycle in question exploits the design in a way that it also

acts whether as a BLSR-like ring, a two-hop segment protecting p -cycle, a full path-segment protecting p -cycle or an NEPC when it comes to prevent node failures.

- In the case of BLSR-like loopback reaction, it only needs to eliminate straddling protection relationships in prior ILPs by setting $\mu_k^{p,r} = 1$ if all three nodes for the two-hop segment under consideration are on the protection structure and zero otherwise.
- From the flow-protecting p -cycle perspective, the parameter $\mu_k^{p,r}$ is now pre-processed in a way defining flow vs. cycle protection-relationships for any path-segment length (i.e., one or more spans to entire working paths, instead of exactly two hops).
- From the NEPC viewpoint, $\mu_k^{p,r} = 2$ for every working route r transiting the node k if and only if p -cycle p is an NEPC for node k ; otherwise $\mu_k^{p,r} = 0$. Because of the scarcity of NEPCs, especially in sparser networks, some paths might not topologically survive certain node failures. ILP₅ is subsequently substituted for ILP₆ in order to address cases where 100% node (and span) restorable designs are not achievable. ILP₆ is similar to ILP₃ but uses the objective function (3.14), which primarily imposes R_{1-node} maximization (rather than 100%) and then requires spare capacity minimization as a second objective.

$$\text{Minimize } \sum_{k \in N} (\Gamma_k - \Theta_k) + \alpha \cdot \sum_{j \in S} C_j \cdot s_j. \quad (3.14)$$

3.2.3 Experimental Results and Discussion

3.2.3.1 Network Characterization

The selection of small and medium size networks given in Section 2.4.4 is used for experiments. The first group of columns in Table 3.2 recalls network characteristics under no-failure conditions. And the 3rd and 4th columns in Table 3.2 report statistics on paths affected by potential node failure events and the number of them that can be considered for restoration, i.e. transiting paths. The ratio of transiting over total paths (including terminating paths which cannot be restored) varies from 25 to 60%.

Figure 3.4 gives a deeper analysis of span versus node failure characterization for the **Havana** network with the traffic matrix of 58 demand-pairs over shortest distance routes—Figure 3.4(a) shows working capacities to be protected against span failures atop edges while histograms in 3.4(b) give equivalent node payloads. In 3.4(b), the x-axis indicates

3. Node-Protecting p -Cycles

Table 3.2: Characteristics of Test Case Networks from the Node Failure Perspective

Networks	N	S	D	P	Affected Paths	Transiting Paths
Havana	17	26	58	135		
	32 nepcs covering 4 nodes.					
<i>(i)initial routing</i>	Shortest distance weighted routing \rightarrow 23,934 working channel-kms.				271	77 (i.e., 28% of affected paths.)
<i>(ii)other routing</i>	Least hop-count based routing \rightarrow 166 working units.				263	69 (i.e., 26% of affected paths.)
<i>(iii)larger traffic matrix</i>	136 D uniformly distributed on [10..100] \rightarrow 2,595,800 working channel-kms.				29,317	14,591 (i.e. 50% of affected paths.)
Cost239	11	26	55	3531	471	119 (i.e., 25% of affected paths.)
	137,170 working channel-kms; 1735 NEPC cover all nodes					
Italy	13	24	78	557	1357	521 (i.e., 38% of affected paths.)
	62,232 working channel-kms. 359 NEPC cover 11 nodes.					
Bellcore	15	28	104	976	1326	396 (i.e., 30% of affected paths.)
	18,335.1 working channel-kms. 847 NEPC cover 8 nodes.					
Euro	32	42	323	699	2483	1497 (i.e., 60% of affected paths.)
	246,375 working channel-kms. 458 NEPC cover 9 nodes.					

possible single node failure scenarios and the blue histogram (i.e., first sets of data) records corresponding numbers of “*affected*” paths. This is the actual number of working paths crossing each specific node in normal network states. Accordingly, *Hanover* and *Frankfurt* have the most impact on the network, as their failures both result in 37 affected paths. In contrast, failures of *Norden* and *Bremen* have the least impact, with 4 affected paths each. Overall, there are 271 combinations of node failures and affected working paths.

The second histogram bars (i.e., red) in Figure 3.4(b) indicate the number of affected paths that are potentially restorable because they are “*transiting*” through a failed node. For example, *Dortmund* failure affects 26 working paths but only 9 of those can be considered for restoration because the 17 others are terminating demands at that node. The blanks in *Berlin*, *Hamburg*, *Munchen* and *Norden* arise because none of the failed paths are transiting those nodes. In totality, 77 failed paths are potentially restorable. The last set of data in Figure 3.4(b), i.e. the yellow histogram, is discussed in the next section.

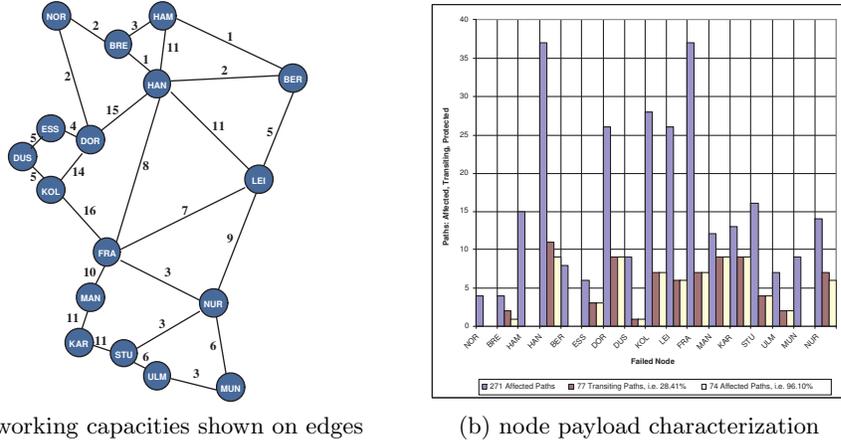


Figure 3.4: Havana Network Characterization from the Node Failure Perspective.

3.2.3.2 Experiments Conducted and Results

A series of experiments was conducted to assess the effectiveness of the proposed “two-hop” node recovery principle. The first set of results correspond to ILP_1 and 8, in which nothing special is done in the design for node failure protection. Test case networks are only planned for 100% span failure restorability (i.e., $R_{1-span} = 1$) at minimum spare capacity. Then, ILP_2 and 10 are used to stimulate node failures and experimentally determine the best R_{1-node} level that can be obtained through two hops and other comparative node recovery methods. Further types of results from ILP_3 and 11 show the level of R_{1-node} that is achievable “for free” under each principle, i.e. with no investment beyond that needed for $R_{1-span} = 1$ only, but free to bias the solution towards choosing cycles that also increase R_{1-node} level. With enhanced ILP_4 and 12, node failure protection is maximized under given extra spare capacity budgets relative to minimum requirements. ILP_5 and 13 were finally used to determine how much spare capacity has to be added to strictly assert 100% R_{1-node} by each method being compared.

Table 3.3 summarizes the experimental results. The second column gives, in terms of channel-kms and spare over working capacity requirements (i.e. redundancy), conventional p -cycle minimum spare capacity solutions for the test case instances under consideration; these are 100% span restorable designs, with no node failure concerns. The third and fourth sets of columns characterize node failure protection aspects, using each of the different

Table 3.3: Sample Results for Current Node Recovery Options using p -Cycles

Networks	Conv. Design spare capa and redundancy (%)		Surviving Paths and R_{1-node} under Min-Costs				Minimum Spare Capacity for 100% (or max.) R_{1-node}			
	2-hop	blsr	npc	flow	fipp	2-hop	blsr	npc	flow	fipp
Havana (i) initial routing	74	60	0	74	77 i.e.	20,444	23,298	11 ie 14% for 28,134	20,335	20,451
	i.e. 96%	i.e. 78%		i.e. 96%	100%	i.e. +0.89%	i.e. +18%	i.e. +39%	i.e. +0.3%	i.e. +0.92%
	53	23	0	53	58 i.e.	186 i.e.	243 i.e.	17 ie 25% for 237	180 i.e.	197 i.e.
(ii) other routing	i.e. 77%	i.e. 33%		i.e. 77%	84%	+39%	+63%	i.e. +77%	+34%	+47%
(iii) larger traffic matrix	12,709	10,766	0	13,177	12,860	3,875,460	4,340,150	1889 i.e.	3,480,250	3,914,447
	i.e. 87%	i.e. 74%		i.e. 90%	i.e. 88%	i.e. +21%	i.e. +36%	26% for 3,550,540	i.e. +9%	i.e. +23%
	i.e. -0.42%							i.e. +11%		
Cost239	92	35	2	94	98 i.e.	99,850	118,420	full for 203,720 ie	98,280	79,705
	i.e. 77%	i.e. 29%	i.e. 2%	i.e. 79%	82%	i.e. +17%	i.e. +38%	+138%	i.e. +15%	i.e. -7%
Italy	468	299	2	468	457	65,696	74,499	359(69%) for 96,518	65,489	73,093
	i.e. 90%	i.e. 57%	i.e. 0.4%	i.e. 90%	i.e. 88%	i.e. +18%	i.e. +34%	i.e. +73%	i.e. +18%	i.e. +31%
Bellcore	338	172	5	338	329	16,199	18,944	135(34%) for 24,329	16,059	14,537
	i.e. 85%	i.e. 43%	i.e. 1%	i.e. 85%	i.e. 83%	i.e. +11%	i.e. +30%	i.e. +67%	i.e. +10%	i.e. -0.37%
Euro	1287	998	83	1350	1272	257,599		415(28%) 388,826	252,715	223,872
	i.e. 86%	i.e. 67%	i.e. 6%	i.e. 90%	i.e. 85%	i.e. +9.52%		i.e. +65%	i.e. +7%	i.e. -5%

restoration options—the 3rd set of columns shows $R_{1\text{-node}}$ levels that are achievable in networks designed for minimum spare capacity but with enhanced $R_{1\text{-node}}$ in mind; and the 4th set of columns reports the amounts of added spare capacity (over min-costs) required to reach 100% restorability against both single span and node failures.

3.2.3.3 Two-Hop Flows vis-à-vis Other Node-Protecting Strategies

In the 3rd and 4th columns of Table 3.3, the first series of data corresponds to the use of two-hop flows. Very high levels of $R_{1\text{-node}}$ (typically 77 to 96%) are achieved under min-costs, and full node restorability is reached for penalties of 0.89 to 21% over minimum spare capacities. The third histogram (i.e., yellow data sets) in Figure 3.4(b) gives a deeper analysis for the *Havana* network considering the original demand matrix. All working paths transiting failed nodes other than *Bremen* and *Hanover* are fully restorable. *Bremen* and the degree-6 *Hanover* are respectively 50 and 82% node restorable. Overall, a total of 74 paths (out of 77) survive single node failure conditions, for a very high level of up to 96% $R_{1\text{-node}}$. This can be pushed to 100%, using the ILP₅ for assertion of 100% $R_{1\text{-node}}$, with less than 1% of additional spare capacity.

The other sets of data in the 3rd and 4th columns of Table 3.3 give comparative results for BLSR-like behavior, NEPC, flow-protecting p -cycle and FIPP. As a generalization of BLSR-like node-protecting principle, the two-hop flow strategy performs better than the simple BLSR-like loopback with $R_{1\text{-node}}$ improvements of 19 to 62% under min-cost conditions and 15 to 63% over extra spare capacity required for full $R_{1\text{-node}}$. Noticeably, the two-hop flow approach is nearly as capacity-efficient as path-segment protecting p -cycles, with a difference of 0 to 20% for both metrics under observation. But significantly flow-segments are otherwise shortened to two-hops in order to guarantee a straightforward failure detection and a real-time activation of right restoration processes.

With NEPCs, $R_{1\text{-node}}$ is almost non-existent under min-costs; it is usually not topologically possible to achieve full $R_{1\text{-node}}$, and it is too costly when possible as with the *Cost239* network. Considering the *Havana* instance with 58 demands over shortest distance routes, the second column of Table 3.2 indicates 32 NEPCs over the 135 distinct simple candidates available across the graph. Those NEPCs cover only four of the 32 nodes, i.e. *Berlin*, *Hamburg*, *Hanover* and *Norden*; and *Hanover* alone was carrying (about 11) transiting paths in

the node payload characterization previously shown in Figure 3.4(b). The 11 working paths correspond to the maximum achievable R_{1-node} of 14%, which requires about 39% extra spare capacity (over min-costs). The usage of non-simple cycles is necessary to achieve an R_{1-node} of 100%, but this is what we are trying to avoid with two-hop flows.

FIPP involves shifting from span to path protection, so the related minimum spare capacity designs differ from that of conventional span-protecting p -cycles, used as benchmark for two-hops, flows and NEPCs. Comparative min-cost solutions in the 2nd column of Table 3.3 seem to suggest that FIPP gives rise to lower costs for instance cases involving more candidate cycles; while conventional p -cycles are better adapted to smaller spaces of candidates. For example, the **Havana** minimum spare capacity FIPP solution is 100% R_{1-node} , but there is an *indirect* penalty as FIPP minimum spare capacity design (i.e., 20,451 channel-kms) is more expensive than that of ordinary p -cycles (i.e., 20,264 channel-kms). As well, FIPP min-cost requirements are even higher than what is required to reach full node restorability using either flows (i.e., 20,335 channel-kms) or two-hops (i.e., 20,444 channel-kms). A detailed comparison between span-protecting p -cycles and FIPPs is provided in Section 3.3.

3.3 A Two-Hop Segment Protecting Paradigm which Unifies Node and Span Failure Recovery under p -Cycles

The previous study on how to derive node failure restorability gave rise to the following question: “If all protected entities within an optical network are segments of exactly two adjacent spans including their common node, is (not) that alone enough to ensure the complete network restorability?” This is referred to as two-hop segment protection and the idea behind is to:

- i. transform any given network topology into another graph where all two-hop segments between nodes are also represented,
- ii. map working routes onto two-hop segments available within the new graph,
- iii. and perform the intended span-like survivable design for the resulting two-hop network.

3.3.1 Two-Hop Protecting p -Cycles

Figure 3.5 illustrates graph transformation and normal state routing within the two-hop graph for the small graph instance in 3.5(a). Two-hop segments $(N_a-N_b-N_c)$, $(N_a-N_c-N_d)$, $(N_a-N_c-N_e)$ and $(N_a-N_d-N_e)$ starting or terminating at node N_a appear in 3.5(b), beside nodes and spans forming the original graph. In 3.5(c), the given working path which follows the span sequence $[(N_a-N_b) (N_b-N_c) (N_c-N_e)]$ is equivalent to the sequence $[(N_a-N_b-N_c) (N_b-N_c-N_e)]$ within the two-hop segment framework. By preventing failures along two-hop segments $(N_a-N_b-N_c)$ and $(N_b-N_c-N_e)$, instead of spans truly crossed by the working route, the given path will be protected against (N_a-N_b) , (N_b-N_c) and (N_c-N_e) span failures and handled under N_b and N_c node failure circumstances.

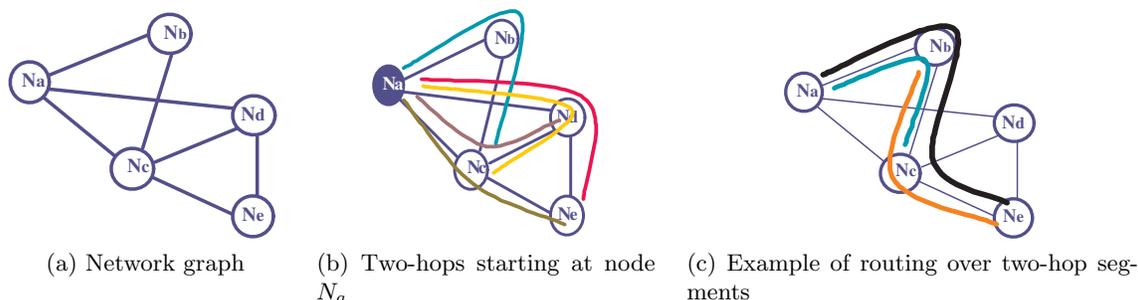


Figure 3.5: *Two-Hop Segments: Graph Transformation and Routing.*

The two-hop segment protection paradigms in the context of p -cycles lead to “two-hop protecting p -cycles”, which are comparable to span-, flow- and path-protecting p -cycles. Figure 3.6 captures the operating principle of two-hop protecting p -cycles. Any two-hop segment having both its end-nodes as part of the same p -cycle structure is entirely restorable within that p -cycle. One or two restoration routes can be available within the p -cycle, depending on whether or not the protection structure also comprises the common node of spans forming the two-hop segment in question. Note that Figure 3.6 is very similar to Figure 3.3; the main difference is that in Figure 3.3, the original p -cycle behavior was applied under span failure circumstances and the two-hop principle was limited to node-protecting purposes (only).

The merit of the two-hop segment protection paradigm is to provide node and span failure recovery within one unique span-like survivability principle. Compared to other

p -cycle architectures, two-hop survivability schemes imply the same simple, fast, and straightforward failure detection and pre-defined switching plans as span protection. Two-hop segments require neither advanced signaling nor centralized management to perform proper restoration actions, contrary to full path-segment oriented architectures such as flow-protecting p -cycles. But two-hop related designs retain the capacity efficiency of full flows-segments, as shown in the experimental results. Unlike end-to-end path-protecting techniques such as FIPPs, two-hop segment protection does not encounter the complexity due to path-disjointness constraints in the design.

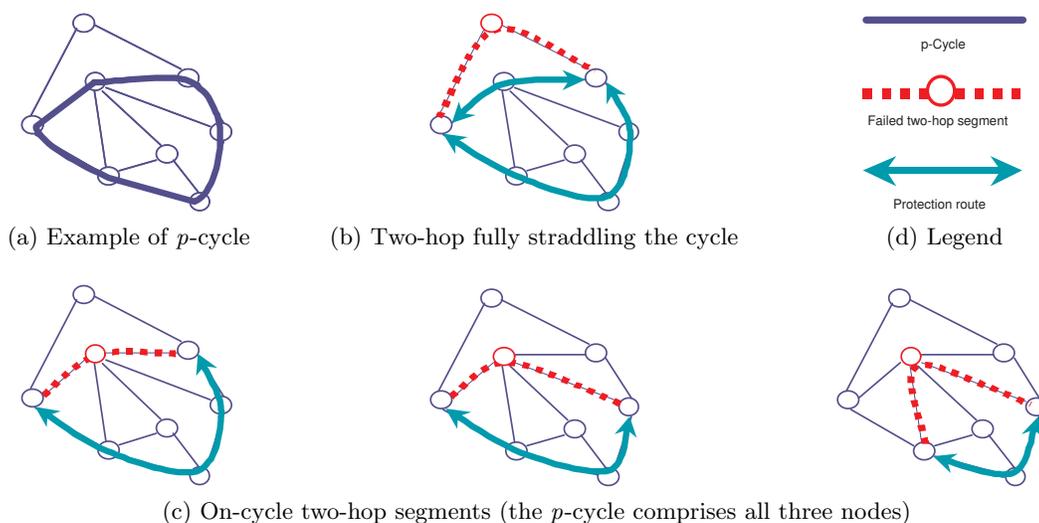


Figure 3.6: Two-Hop Segment Protection using p -Cycles.

3.3.2 An ILP Design Model for Two-Hop Protecting p -Cycles

This section formulates ILP₇ mathematical design model for two-hop protecting p -cycles. These are several points to keep in mind for an effective design of two-hop protecting p -cycles. The working routes of one single span do not match the two-hop segment paradigm. It is certainly possible to require an initial routing where every path would be at least two hops, but the present ILP formulation addresses the more general case of any path hop-length. On the other hand, two distinct two-hop segments can use the same p -cycle copy if and only if they have no common span and they do not share the same intermediate node. This is because a single span (or node) failure implies the simultaneous failure of

all two-hop segments comprising the span (or the intermediate node) in question. The following definitions serve for ILP₇ formulation.

Sets Besides S , N , D and P , another set H is required to represent two-hop segments available within a graph. H is indexed by a pair of adjacent spans (i, j) .

Input Parameters In addition to prior input parameters d^r , C_j and x_i^p ,

- a replication, $y_{(i,j)}$, of x_i^p defines protection relationships between two-hop segments and p -cycles. Accordingly, $y_{(i,j)}^p = 2$ if the two-hop segment formed by spans i and j fully straddles p ; $y_{(i,j)}^p = 1$ if p comprises all three nodes involved in the two-hop segment (i, j) ; otherwise, $y_{(i,j)}^p = 0$.
- $\delta_j^r \in \{0, 1\}$ indicates spans that each given demand relation r crosses en route ($\delta_j^r = 1$ if r crosses j and $\delta_j^r = 0$ otherwise). This serves for mapping of working routes onto the two-hop (transformed) graph.
- $\mu_k^j \in \{0, 1\}$ indicates whether or not node k is an end-node for span j ; ($\mu_k^j = 1$ if it is, and $\mu_k^j = 0$ if it is not).

Decision Variables s_j and η^p remain as before while

- n_i^p records the number of unit-sized copies of p -cycle p handling one-hop paths carried on span i in the design.
- $m_{(i,j)}^p$ similarly records the number of unit-sized copies of p -cycle p assigned to the two-hop segment (i, j) in the design.

ILP₇ Formulation — Equations (3.15)-(3.20) define ILP₇ for the design of two-hop protecting p -cycles. The objective function (3.15) minimizes total spare capacity required in the design. The 100% two-hop restorability constraint (3.16) prevents both single node and span failures along any two-hop segment. Equation (3.17) extends full restorability requirements to spans that comprise single-hop paths. Equation (3.18) makes a sufficient provision of p -cycles to handle the sum of all one-hop paths *plus* two-hop segments crossing each given span. With equation (3.19), the same purpose as in (3.18) is achieved but for

any node failure scenario and two-hop segments transiting the node in question. Equation (3.20) performs spare capacity placement in regard to p -cycles selected in the solution.

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j. \quad (3.15)$$

$$\sum_{r \in D: \delta_i^r=1 \text{ and } \delta_j^r=1} d^r \leq \sum_{p \in P} y_{(i,j)}^p \cdot m_{(i,j)}^p, \forall (i,j) \in H. \quad (3.16)$$

$$\sum_{r \in D: \sum_{j \in S} \delta_j^r=1} d^r \cdot \delta_i^r \leq \sum_{p \in P} x_i^p \cdot n_i^p, \forall i \in S. \quad (3.17)$$

$$n_i^p + \sum_{j \in S: (i,j) \in H} m_{(i,j)}^p + \sum_{j \in S: (j,i) \in H} m_{(j,i)}^p \leq \eta^p, \forall p \in P, \forall i \in S. \quad (3.18)$$

$$\sum_{(i,j) \in H: \mu_k^i=1 \text{ and } \mu_k^j=1} m_{(i,j)}^p \leq \eta^p, \forall p \in P, \forall k \in N. \quad (3.19)$$

$$s_j = \sum_{p \in P: x_j^p=1} \eta^p, \forall j \in S. \quad (3.20)$$

3.3.3 Two-Hop Protecting p -Cycles vis-à-vis Other p -Cycle Schemes

A series of experiments was conducted by applying ILP₇ to the collection of test case networks. Table 3.4 records sample results: the first column recalls test case instances and the second set of columns characterizes them from the perspectives of hop-lengths of working paths and average nodal degrees. Column 3 records total spare channel-kms for two-hop protecting p -cycles. For comparison purposes, the fourth and fifth sets of columns give equivalent results for span- and path-protecting p -cycles, which results were previously generated as part of the study in Section 3.2.

3.3.3.1 Span- versus Path-Protecting p -Cycles

From the beginning, FIPP p -cycles have been suspected to be more capacity efficient than ordinary span-protecting p -cycles. A true analysis is now possible by comparing network solutions in the 4th and 5th columns of Table 3.4. The fourth set of columns pertains to span-protecting p -cycle solutions while the fifth set of columns relates to FIPP-based designs. In both columns, the first set of data records total spare capacity required for

3. Node-Protecting p -Cycles

100% span restorability with no node failure concerns, and the second data set indicates what is required to achieve full protection against both span and node failures.

Table 3.4: Sample Results for Comparative Study of p -Cycle Related Architectures

Networks	Initial Routing		Two-Hop Protecting p -Cycles	Span-Protecting p -Cycles		Path-Protecting p -Cycles	
	path-hops	node degree		R_{1-span} only	full R_{1-node}	R_{1-span} only	full R_{1-node}
Havana							
(i) original	1.79	3.06	19,523 i.e. 81.57%	20,264 i.e. +3.10%	20,444 i.e. +3.85%	20,451 i.e. +3.88%	20,451 i.e. +3.88%
(ii) hop-count routing	1.71		121 i.e. 72.89%	134 i.e. +7.83%	186 i.e. +39.16%	157 i.e. +21.69%	197 i.e. +45.78%
(iii) larger traffic matrix	2.98		2,713,684 i.e. 104.54%	3,187,827 i.e. +18.27%	3,875,460 i.e. +44.76%	3,174,532 i.e. +17.75%	3,914,447 i.e. +46.26%
Cost239	1.67	4.73	41,790 i.e. 30.47%	85,640 i.e. +31.97%	99,850 i.e. +42.33%	75,970 i.e. +24.92%	79,705 i.e. +27.64%
Italy	2.24	3.69	36,505 i.e. 58.56%	55,654 i.e. +30.72%	65,696 i.e. +46.83%	47,735 i.e. +18.01%	73,093 i.e. +58.69%
Bellcore	1.85	3.73	9485 i.e. 51.73%	14,591 i.e. +27.85%	16,199 i.e. +36.62%	13,808 i.e. +23.58%	14,537 i.e. +27.55%
Euro	4.036	2.63	291,038 i.e. 118.13%	235,206 i.e. -22.66%	257,599 i.e. -13.57%	213,532 i.e. -31.46%	223,872 i.e. -27.26%

- i. In contrast to a priori thoughts on FIPP capacity-efficiency, sample results suggest that conventional span-protecting p -cycles are more capacity-efficient than path-based p -cycle architectures for very sparse and lightly loaded networks. This is the case for **Havana** instances when using the original demand matrix, where improvements are of 0.92% and 17.16% for distance-weighted and hop-count working routings.
- ii. FIPP yields slightly better results in medium and highly loaded sparse networks. For example, an improvement of 0.41% was obtained with FIPP relative to span-protecting p -cycle solution for the **Havana** network under a high demand load.
- iii. The efficiency of path-protecting p -cycles over conventional span-protecting p -cycles is more perceptible in dense networks with wide spaces of candidate cycles. Improvements over regular span-protecting p -cycle solutions are of 11.32% in the **Cost239**, 13.08% the **Italian** and 5.37% in the **Bellcore** networks.

- iv. In Euro instance which is both dense and highly loaded, the gain of FIPP p -cycles over conventional span-protecting p -cycles is 9.21%.

3.3.3.2 Capacity-Efficiency of Two-Hop Protecting p -Cycles

In contrast to span- and path-based p -cycles, two-hop protecting p -cycles directly achieve both $R_{1\text{-span}} = 1$ and $R_{1\text{-node}} = 1$. And two-hop protecting p -cycles are typically much more capacity-efficient than any other p -cycle variant. Improvements over conventional span-protecting p -cycles reach up to 32% when considering full $R_{1\text{-span}}$ alone and 47% when full $R_{1\text{-node}}$ is required in addition; equivalences for path-protecting p -cycles are 25 and 59%, respectively. In Euro network, the trend for capacity efficiency of two-hop protecting p -cycles over other methods is reversed. Deterioration vis-à-vis conventional span-protecting p -cycles is 23% for $R_{1\text{-span}}$ alone and 14% when adding the full $R_{1\text{-node}}$ constraint; vis-à-vis path-protecting p -cycles, differences are 32% for $R_{1\text{-span}}$ alone and 27% for combined $R_{1\text{-node}}$ and $R_{1\text{-span}}$. What is wrong with the Euro network?

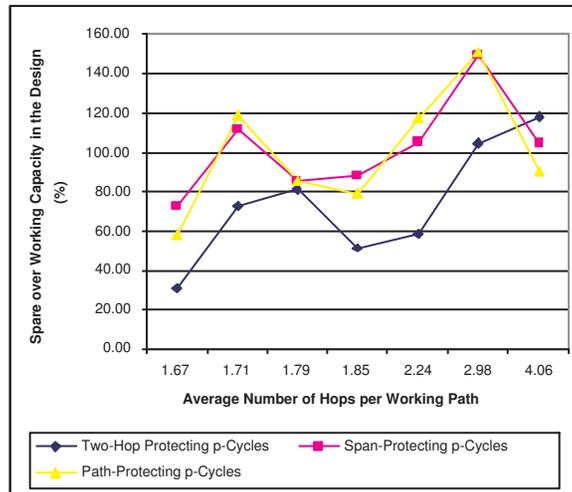


Figure 3.7: Impact of Path Hop-counts on the Performance of Two-Hop Protecting p -Cycles.

To handle intermediate node failures, two-hop protecting p -cycles require two times the protection needed from the 2nd to the $(n - 1)$ th spans along any considered path of n hops. So first doubts about the questioning performance for Euro network pertain to the average hop-length of working paths. Accordingly, the x-axis in Figure 3.7 ranks test case instances

3. Node-Protecting p -Cycles

by ascending order of average path hop-lengths. For each network instance, the y-axis indicates spare over working capacity ratios in the design under three plots pertaining to the p -cycle methods under consideration. In Figure 3.7, gaps between the "blue" curve for two-hop protecting p -cycles and plots for span- and path-protecting p -cycles do not follow a particular behavior; so path hop-lengths do not justify the questioning performance of two-hop protecting p -cycles for Euro.

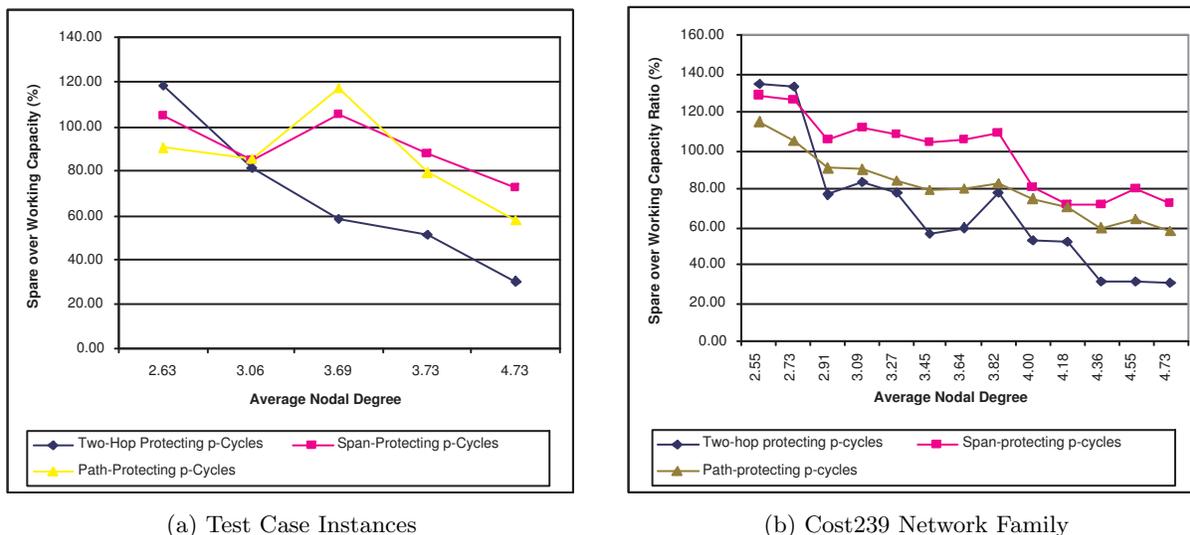


Figure 3.8: Impact of Average Nodal Degree on the Performance of Two-Hop Protecting p -Cycles.

The number of two-hop segments available within each test case instance or, more generally, network average nodal degrees may also explain the disappointing performance of two-hop protecting p -cycles in Euro network. Figure 3.8 studies nodal degrees versus capacity redundancies for two-hop, span- and path-protecting p -cycles. Two sets of networks are under observation: Havana, Cost239, Italy, Bellcore and Euro master cases; and a Cost239 network family obtained by using the original Cost239 instance as master topology and then randomly removing spans, in order to decrease the network nodal degree while preserving its bi-connectivity. Thirteen networks of 11 nodes and 14 to 26 spans, for average nodal degrees of 2.55 to 4.73, were derived from the Cost239 network. (The original set of demands is still considered.) In Figure 3.8, capacity redundancies of two-hop protecting p -cycles are always below that of other approaches for degree-3 networks, mean-

ing that efficiency of two-hop protecting p -cycle designs is only questionable for network instances of less than degree-3 (such as the Euro network). On the other hand, the gap for two-hop efficiency over span- and path-protecting p -cycles improves with the growth of network average nodal degrees.

3.3.4 Summary of Two-Hop Protecting p -Cycle Features

Table 3.5 summarizes the benefits of two-hop protecting p -cycles, which show fairly high competitiveness over other p -cycle survivability schemes in terms of capacity-efficiency in the design, simplicity and proper switching operations, non-complexity in the design and intrinsic capabilities for full node failure recovery. All of these may justify further research from such perspectives as dual span failure protection, length control in the restored network state and multiple quality-of-protection. In a more general manner, two-hop segment protection is extendable to any other span-like survivability scheme.

Table 3.5: *Benefits of Two-Hop Protecting p -Cycles over of Other Related Schemes*

Protected entities	Spans	Two-hop segments	Full flow-segments	End-to-end paths
Spare capacity requirements	low	less than in spans and paths from degree-3	least possible	low
Operations	simple, fast, easy, straightforward failure detection, pre-defined switching plans, etc.		centralized management	same as spans and two-hops
Design complexity	relatively simple to design			too much complexity
R_{1-node}	through extensions such as 2-hop flows	inherently integrated	require node-disjointness constraints in the design	
Others	R _{2-span} , optical reach control, multi-QoP, etc.			

3.4 Near-Optimal FIPP Network Designs using General Path-Protecting p -Cycles

A typical FIPP network comprises several p -cycles of which each provides protection paths to a set of working routes. Each cycle and its protected route set are collectively referred to as a *configuration*. The two types of FIPP configurations shown in Figure 3.9 are typically considered in the literature.

- i. The first type, which will be referred to as **Type 1**, corresponds to a situation where only the following condition is permitted: the working routes protected in a given FIPP configuration must be mutually span disjoint so that they cannot be simultaneously affected by the same single span failure. The route set protected as part of such a configuration is referred to a *disjoint route set* (DRS).
- ii. The second type of FIPP configurations, which will be referred to as **Type 2**, corresponds to the case where both conditions i. and ii. are permitted. Condition ii: if the working routes are not mutually span disjoint, their protection paths must be.



(i) FIPP protecting a set of span disjoint working routes (ii) Non-disjoint paths sharing the same FIPP

Figure 3.9: FIPP Configuration Types: Type₁-condition (i); Type₂-both conditions (i) and (ii).

3.4.1 Current FIPP Design Methods

In an effort to reduce the overall complexity of the FIPP design problem, most of the methods presented in the FIPP p -cycle literature (e.g. FIPP-ILP, FIPP-DRS) generate designs composed entirely of **Type 1** configuration. The FIPP-CG approach is the one that considers the most general FIPP p -cycle case, meaning **Type 2** configuration.

FIPP ILP The original method to generate FIPP designs is found in [GK05, KG05a]. This is an ILP requiring the pre-processing of shortest working routes, candidate cycles, and *rival demand-pairs* which cannot share the same p -cycles. The FIPP ILP method disallows a single demand bundle to be protected by distinct cycle structures, despite that it is optimal to do so. Also, the original FIPP ILP method is very difficult to solve for anything but small-sized networks, because of rivalry-integrated constraints. For these reasons, the original FIPP ILP is not featured in this thesis.

FIPP DRS The FIPP DRS method was introduced in [KGD05] as a practical method for solving FIPP p -cycle problems for medium sized networks. FIPP DRS works by pre-generating **Type 1** configurations in a way that ensures an adequate representativeness of every demand. The resulting set of configurations is then passed to an ILP, the use of which optimally determines the lowest cost set of configurations that provides 100% single span failure restorability for the entire demand set. The use of FIPP DRS method generates optimal FIPP p -cycle designs only if configurations found in the optimal solution are given to the ILP as part of a pre-generated configuration set. Typically only a small set of configurations is generated relative to the set of all possible configurations. Thus, the use of FIPP DRS yields sub-optimal but still fairly efficient FIPP p -cycle designs. FIPP DRS is the most common FIPP design method found within the literature. In this research, the FIPP DRS approach is also considered for comparison purposes.

FIPP CG [JRBG07] uses the technique of column generation to generate optimal or near optimal FIPP p -cycle designs matching **Type 2** configuration. The idea behind FIPP CG is to decompose FIPP p -cycle design problem instances into master and pricing sub-problems. The master sub-problem is used iteratively to minimize the total cost of the design while considering a small set of configurations (at a given time). New configurations are generated every iteration by the pricing sub-problem and added to the set considered by the master problem only if they can reduce the overall cost of the design. If no such cost-improving configuration is found by the pricing problem, then the most recent solution obtained by the master problem is provably within 1% of optimality.

FIPP CG implementation is more complicated than that of other FIPP methods and requires a fairly good initial feasible solution (in itself not a trivial task) to terminate in a

reasonable amount of time. This method is not formulated to generate optimal solutions that utilize **Type 1** configurations as in this research. However, the use of FIPP CG related results will allow to investigate the effect that removing the working route disjointness constraint has on the amount of spare capacity required in optimal FIPP p -cycle solutions.

3.4.2 General Path-Protecting (GPP) p -Cycles

3.4.2.1 FIPP vs. GPP

Nearly all FIPP design approaches explicitly build configurations as part of an ILP. This results in a large number of binary variables which significantly increases the complexity of the problem and/or results in very long solver run times. Efforts to eliminate the exceeding variables in earlier FIPP models generalized the original FIPP definition within the concept of *general path-protecting (GPP) p -cycles*. The main difference is that under regular FIPPs, individual working paths are constrained to be protected by a single cycle structure as part of a single configuration; while under GPPs, unit working paths are allowed to switch onto different p -cycles from one failure to another.

Figure 3.10 captures the GPP idea: in 3.10(a) and 3.10(b), working paths D_1 and D_2 are respectively protected by unit p -cycles P_{1-1} and P_{1-2} . Under GPPs, path D_3 in 3.10(c) can be restored using prior P_{1-1} if span B fails, and P_{1-2} if span A fails. Thus, 2 cycles are required to achieve a 100% single span failure restorable design using GPPs, while the use of FIPP p -cycles requires to create a third configuration for D_3 .

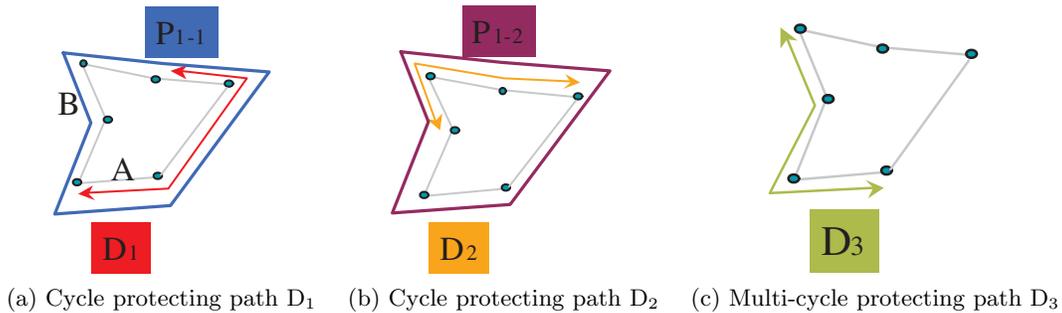


Figure 3.10: *Illustration of the GPP Concept.*

From another perspective, the protection paths provided by P_{1-1} and P_{1-2} for D_3 are routed identically in this example, as they follow the same path-segment of the same cycle

structure. But this does not have to be the case in general, meaning that routes for the protection paths used to restore D_3 could be completely different from one span failure to another. Thus, GPP requires more failure localization and additional signaling than FIPP to take the correct restoration actions when working paths use *multi-cycle* protection.

3.4.2.2 ILP₈ for GPPs

GPP optimal solutions are much more accessible than FIPP ones. The concept of GPPs can be very efficiently captured into an ILP mathematical design model, quite similar to ILP₇ for two-hop protecting p -cycles.

- In ILP₈ for GPPs, ILP₇ parameter $y_{(i,j)}^r$ just becomes y_r^p ; $y_r^p \in \{0, 1, 2\}$ and encodes the number of protection segments that one unit-sized copy of p -cycle p may provide to the demand relation r : $y_r^p = 2$ if working route for r fully straddles p -cycle p , $y_r^p = 1$ if r is in a full or partial on-cycle relationship with p , and $y_r^p = 0$ otherwise.
- Prior decision variable $m_{(i,j)}^p$ similarly becomes m_r^p to represent the number of copies of p -cycle p assigned to the demand relation r in the design.

Equations (3.21)-(3.24) define ILP₈ for full span restorable GPP network solutions. The objective function (3.21) optimizes total spare capacity in the design. Equation (3.22) ensures that 100% of traffic flows will survive any single span failure, and equation (3.23) selects and capacitates cycle structures for this purpose while ensuring that enough cycle copies are available in the design to handle rival paths with one or more spans in common. Equation (3.24) places spare capacities on each specific edge.

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j. \quad (3.21)$$

$$\sum_{p \in P} y_r^p \cdot m_r^p \geq d^r, \quad \forall r \in D. \quad (3.22)$$

$$\sum_{r \in D} m_r^p \cdot \delta_i^r \leq \eta^p, \quad \forall i \in S, \quad \forall p \in P. \quad (3.23)$$

$$s_j = \sum_{p \in P: x_j^p=1} \eta^p, \quad \forall j \in S. \quad (3.24)$$

3.4.3 Deriving a FIPP Solution from an Existing GPP Design

The GPP closeness to FIPP and the simplicity of GPPs in the design may lead a three-step approach to obtain efficient FIPP solutions.

- i. ILP₈ is first used to generate an optimal GPP network design;
- ii. p -cycles involved in the GPP design solution are uniquely assigned to working paths under FIPP DRS constraints, while minimizing the number of unprotected paths;
- iii. shortest p -cycles available in the prior GPP solution, which have protection relationships with unprotected demands, are duplicated to obtain an effective FIPP design.

3.4.3.1 Characterizing GPP Designs from the FIPP Perspective using ILP₉

ILP₉, defined by equations (3.25) to (3.28), optimally assigns each working path to a unit cycle while minimizing the number of paths that remain unprotected. This assumes an existing GPP solution where each single unit p -cycle is recorded in a set SP ; if the GPP design contains three copies of cycle p_1 , the set SP will also contain three distinct unit entries (p_{1-1} , p_{1-2} and p_{1-3}) for p_1 . Binary variable ψ_r^p assigns unit cycles p to demand relations r while decision variable v^r counts unprotected demands. The objective function (3.25) minimizes unprotected demands in the design. Equation (3.26) assigns unit p -cycles to working paths. Constraint (3.27) guarantees that any working path set protected by the same p -cycles respect span disjointness requirements. Equation (3.28) makes sure that unit p -cycles are only assigned to working paths they are eligible to protect.

$$\text{Minimize } \sum_{r \in D} v^r. \quad (3.25)$$

$$\sum_{p \in SP} y_r^p \cdot \psi_r^p + v^r \geq d^r, \quad \forall r \in D. \quad (3.26)$$

$$\sum_{r \in D} \psi_r^p \cdot \delta_j^r \geq 1, \quad \forall p \in SP, \quad \forall j \in S. \quad (3.27)$$

$$\psi_r^p \leq y_r^p, \quad \forall r \in D, \quad \forall p \in SP. \quad (3.28)$$

3.4.3.2 Extracting a Final FIPP Solution

In ILP₉, unprotected paths essentially identify multi-cycle protected paths in any given GPP network design. If ILP₉ reveals no unprotected path, the optimal GPP solution also corresponds to an optimal FIPP design. Otherwise, to convert that GPP design into a FIPP solution, unprotected paths are considered in turn; one copy of the shortest p -cycle, available within the GPP design in question, which can protect the unprotected path under consideration is added to the solution.

3.4.4 Effectiveness of FIPP Design Solutions through GPPs

The series of experiments consists of producing optimal GPP designs and then converting them into FIPP solutions. Table 3.6 reports experimental results: the first column lists the test case networks and the second set of columns records spare capacity requirements for GPP optimal designs and for FIPP p -cycle solutions extracted from those GPP designs, *plus* the number of working paths that remain unprotected when characterizing GPP design solutions from the FIPP perspective. The third set of columns is for comparison with DRS and CG design approaches. Solutions were generated as part of previous research in [JRBG07, GGC⁺07] by Dimitri Baloukov, a colleague from TRILabs and the University of Alberta, Professor Brigitte Jaumard from Concordia University in Montréal, and her former PhD student Caroline Rocha. As DRS and CG provided solutions do not specifically relate to this PhD research project, results only cover **Havana** (when considering the original set of demands) and **Cost239** networks rather than all test case instances.

In Table 3.6, imposing the constraint of failure independence to a GPP design solution results in 0 or 1 unprotected path for test case instances, meaning that GPP solutions also stand as optimal or near-optimal FIPP network solutions. Furthermore, FIPP DRS solutions are respectively $\sim 8\%$, $\sim 12\%$ and $\sim 23\%$ less efficient than that of FIPP GPP in each provided solution. But FIPP CG solutions, which are also optimal, result in $\sim 6\%$ and $\sim 9\%$ lower spare capacity cost than the solutions obtained using GPPs. The reason is that GPP and DRS methods only consider configurations of **Type 1** while CG-based solutions consider **Type 2** configurations, which are a more general case. This result is particularly important because comparing near-optimal FIPP solutions with optimal CG solutions gives insight into how much spare capacity can be saved by relaxing the working route disjointness

constraint in FIPP designs. This insight was not available before because methods for generating FIPP solutions of both types did not cover any network and demand size.

Table 3.6: *Sample Results for FIPP Design Solutions through GPP*

Networks	GPP	FIPP-derived Design	DRS	CG
Havana				
(i) <i>original</i>	20,451	20,451 (0 unprotected path)	22,056 i.e. +7.84%	
(ii) <i>hop-count routing</i>	157	157 (0 unprotected path)	176 i.e. +12.10%	147 i.e. -6.37%
(iii) <i>larger traffic matrix</i>	3,174,532	3,176,110 (1 unprotected path)		
Cost239	75,970	75,970 (0 unprotected path)	93,345 i.e. +22.87%	68,840 i.e. -9.38%
Italy	47,735	48,373 (1 unprotected path)		
Bellcore	13,808	13,808 (0 unprotected path)		
Euro	213,532	213,532 (0 unprotected path)		

3.4.5 Node Failure Protection using FIPPs through GPPs

Table 3.1 describes ILP_{10–13} as GPP equivalences of ILP_{2–5} for two-hop, full flow and NEPC node-protecting p -cycle strategies. The underlying principle within ILP_{10–13} is to add equations (3.29)-(3.31) to the original GPP-ILP constraints (3.22)-(3.23), while not changing equations (3.4)-(3.13). In addition to span-disjointness needs, equation (3.29) requires node-disjointness from working paths sharing p -cycles. Equation (3.30) assigns protection segments, within available p -cycles, to working paths and equation (3.31) calculates cycle copies required in the design.

$$\sum_{r \in D} m_r^p \cdot \varepsilon_k^r \leq \eta^p, \quad \forall k \in N, \quad \forall p \in P. \quad (3.29)$$

$$\theta_k^{p,r} \leq n_k^{p,r} \cdot y_r^p, \quad \forall r \in D, \quad \forall k \in N, \quad \forall p \in P. \quad (3.30)$$

$$n_k^{p,r} \leq m_r^p, \quad \forall r \in D, \quad \forall k \in N, \quad \forall p \in P. \quad (3.31)$$

To obtain effective node-protecting FIPP design solutions, it is possible to impose span and node failure independence constraints to optimal GPP design solutions and duplicate

(if needed) shortest p -cycles capable of handling unprotected working paths. But in this chapter, GPP (span and node) restorable designs were directly considered as FIPP solutions for simplicity because GPP related designs are now proven to be very close to that of FIPPs.

3.5 Concluding Discussion

This chapter has revealed a new, relatively simple and possibly cost-effective approach to achieve combined protection of optical networks against both node and span failures. The new insight is based on a generalization of how nodes in a BLSR-ring or p -cycle (to date) derive survivability through loopback at the nearest two neighbor-nodes on the same ring. The generalization views any combination of node failure and an affected transiting path from the standpoint of the two-hop segment defined by the failed node and the nodes immediately adjacent on the affected path. It does not matter if these nodes are found together within the same p -cycle as the failed node, or another p -cycle entirely; the transiting path affected by the node failure is inherently restorable by ordinary p -cycle switching actions whether the respective two-hop segment is on-cycle, straddling, or partially on-cycle and partially straddling. Resulting designs for enhanced R_{1-node} under controlled penalties and strict assertion of $R_{1-node} = 1$ use only slightly more capacity than corresponding optimal sets of p -cycles for $R_{1-span} = 1$ only. More generally, the two-hop approach for node-protecting p -cycles appears to be a good compromise with related concepts.

Two-hop protecting p -cycles were also introduced and explored with the aim of protecting segments of exactly two-hops within an optical network, in order to prevent both span and node failure events using one unique principle. The study explained how to transform a given network topology and its demand working routes from the two-hop perspective and then perform a p -cycle design for the obtained two-hop graph. In the experiments, two-hop protecting p -cycles appeared as a promising alternative to conventional span-, flow-protecting and FIPP p -cycles. Two-hop protecting p -cycles were especially capacity-efficient, relative to other p -cycle architectures under comparison, in degree-3 or more network instances. On the basis of this performance, the two-hop segment protection option can be considered as a completely new paradigm besides other span- and path-based survivability schemes.

This chapter also introduced a novel strategy for obtaining FIPP p -cycle solutions on the basis of GPP designs in which the constraint of failure independence is first relaxed. The merit of the proposed approach is that GPPs use a much simpler mathematical formulation than what is required to generate FIPP p -cycle solutions directly. By imposing the failure independence constraint onto GPP solutions and identifying resulting unprotected working paths, which never exceed two, FIPP p -cycle solutions were extracted by capacitating additional cycles able to protect these paths. FIPP solutions obtained through GPPs were considerably better (by as much as 23%) than those obtained by the widely used FIPP DRS method. Results also showed that relaxing the disjoint route set constraint in FIPP p -cycle networks can result in as much as a 9% decrease in spare capacity cost. This study also ventured to provide a true comparison of span-protecting p -cycles with FIPPs from the capacity efficiency perspective. Previously, this had not been possible because prior FIPP design methods were too difficult to solve for many real-size networks.

Chapter 4

Fully Transparent p -Cycle Designs

This chapter is adapted and extended from [OG08a, OG09c, OG11, GGC⁺09], which address several transparency issues in p -cycle related networks.

- One contribution is the direct control of the combined length of working paths *plus* protection path-segments they might use in the restored network states, through an intelligent matching of working paths crossing a failed span with shorter protection path-segments available within p -cycles involved in the solution and showing protecting-relationships with the span in question. Taken overall, this complementary matching principle provides a means to design an entire transparent survivable island that respects the transparent reach limits of a given ultra long-haul technology.
- Other contributions comprise a standardized CapEx cost model for various p -cycle WDM configuration types, and an exploratory study of *whole fiber switched p-cycles* as a promising alternative to the prior p -cycle configurations in WDM networks (referred to as *wavelength switched p-cycles in this thesis*). We discuss the practicability of glass switched p -cycles from a technological viewpoint, highlight enhancements from the design complexity and real-world CapEx cost perspectives, and mention possible applications.

The chapter is outlined as follows. Section 4.1 introduces the novel matching strategy that intelligently assign working paths crossing a failed span with shorter protection segments through available p -cycles, thereby controlling restored state path lengths in fully transparent p -cycle networks. Section 4.2 reports the exploratory study of whole fiber switched p -cycles, for the purpose of reducing the cost of fiber selective ports in a cross-connect switch while removing the complexity due to wavelength continuity concerns in prior-known fully transparent p -cycle configurations. Section 4.3 concludes the chapter.

4.1 p -Cycle Design with Controlled Optical Path Lengths in the Restored Network State

Lightpath signals undergo degradation when traversing a transparent optical domain due to factors primarily associated with the length [and hence loss and other accumulated impairments affecting the signal to noise ratio (SNR)] of optical paths. To ensure a signal with adequate post-detection bit error rate (BER) levels (-10^{-12} at least), an optical signal may eventually require regeneration along its path. However, regeneration is costly because it involves electro-optical conversions. So it is generally desirable to try to avoid, or minimize, the need to regenerate signals en route during the network design phase.

A networking approach related to this issue is the concept of *translucent* networks. A translucent network consists of optically transparent domains or islands interconnected with electro-optical switches having regeneration capabilities. The idea is to confine signal regeneration functionalities at the boundaries of islands of transparency, instead of requiring regeneration capability at intermediate nodes within transparent domains [EGRG05]. This constrains, in a specific transparent domain, optical lightpaths to be under the maximum length that signal can travel before the SNR ratio degenerates unacceptably. A typical maximum transparent distance of currently sold ultra-long-haul (ULH) dense WDM systems is reported in [CWN⁺02] to be about 4000 km.

Routing lightpaths over the shortest routes is an obvious way to reduce the length of working paths in normal operations. Defining length-limited *protection paths* can take more consideration, depending on the self-healing architecture implemented in the survivable network. In the case of path-protecting architectures, it is still simple to judge if a restored state path length is under the limit or not because protection paths completely replace the failed working paths, end-to-end. Considering for example 1+1 APS [BRS04], the longer half of the smallest circumference-size cycle joining the two end-nodes of a given demand-pair would usually be considered as the restoration path, and the shorter side used for the working route. Then, backup is the next-shortest route fully disjoint from the primary route, and it is simple to see if a length limitation is met or not in the restored state (although for 1+1 APS, the protection path length may be high due to the full path disjointness requirement, especially in sparser networks). But controlling end-to-end optical

path lengths in restored network states is not so simple with span-protecting architectures because the end-to-end path length will depend, in the event of span failure, on both the length of the protection path-segment and the lengths of non-failed portions along the affected working path.

This research considers methods that can efficiently control the length of end-to-end restored state paths in the design process of span-protecting p -cycles. In opaque p -cycle designs, where wavelength conversion is assumed at all nodes, the length of protection lightpaths is not of primary concern because regeneration happens at every node. Thus, in the most commonly used basic design model for minimum spare capacity (i.e. ILP₁), we just need to ensure that there is enough p -cycle capacity available to protect all the failed working channels on any span failure. The routing and length of pre-failure working paths do not even appear in the problem. That information is, in effect, just boiled down to the number of working channels on each span. In transparent optical networks, the basic p -cycle design problem is of greater complexity because one must also not exceed specific optical path length limits.

4.1.1 Prior Related Work

In earlier work, optical restored state path lengths were indirectly limited or reduced by using only circumference-limited candidate cycles, instead of considering all possible cycles as eligible in the design [SGA02, Gro03b]. This works if there is some upper limit on cycle circumference size that can be stated independently of any working path length such that the combined maximum lengths will still be adequate. But that condition may cause many paths to be shorter than needed or the diameter of a transparent domain to be smaller than it otherwise might be. This principle is also applicable to hybrid configuration types, in which working paths and p -cycles are independently optically transparent but there is regeneration required, specifically located at the entry points of working paths into p -cycles.

An improvement over just limiting candidate cycle sizes is found in [KSG05], which recognizes that with respect to any given protected span, a p -cycle may offer a “longer side” and a “shorter side,” and we may preferably choose the short side or exclude any use of the long side if it is too long. But as in [SGA02, Gro03b], the authors in [KSG05] do not consider the affected working paths’ original length in the design. This approach can

minimize or strictly cap the maximum length of protection path-segments without directly constraining the circumference of candidate p -cycles used in the design. Although the work in [KSG05] distinguishes between the long and short sides of a p -cycle with respect to prospective span failures, the method involves simply not using any protection path-segment that in its own right exceeded a length limit. There is no consideration or direct control over the combined length of working paths *plus* protection path-segments they might use.

The preliminary study in [OG07] constitutes a somewhat related work. [OG07] is aimed at showing the possible flexibility in terms of possible operator objectives, of using only small p -cycles or using the fewest number of p -cycles in the design. But none of [SGA02, Gro03b, KSG05, OG07] introduces methods for direct control of end-to-end optical path lengths in the restored network state.

4.1.2 Matching Longer Working Paths with Shorter Path-Segments in the Available p -Cycles

This study explores p -cycle network design with the purpose of reducing the length of end-to-end lightpaths in restored states, when a span failure occurs. The basic strategy consists of systematically matching longer working paths crossing a failed span with shorter path-segments within the p -cycles available to protect that span. Ideally, the aim is to thereby control the end-to-end length of all paths in restored network states, as part of a new p -cycle minimum spare capacity design model. The main technical difference over prior studies is that now, in the design problem, p -cycles are chosen and the related failure protection preplanning information is produced while taking into consideration the original lengths of working paths. The novel matching strategy does not limit candidate cycle lengths with the advantage that the set of all possible cycles is still considered in the design, leading to the most capacity-efficient solutions. Also, there is a globally optimized control of the end-to-end optical lightpaths in the restored state, not just control of the length of a maximum protection path-segment. This stands in contrast to previous attempts to reduce or limit optical path lengths in the restored state, which have been based on the more indirect approach of limiting the circumference of candidate cycles [SGA02] or the length

4. Fully Transparent p -Cycle Designs

of segments used within cycles [KSG05], without taking information about the original lengths of working paths into account as is done now.

Figure 4.1 illustrates the idea of matching longer working lightpaths with shorter protection path-segments through available p -cycles. In 4.1(a), two p -cycles protect against failures on the span *Zurich to Prague* (considering the *Cost239* pan-European network). The longer cycle is in a so-called “on-cycle” relationship to the protected span and thus provides one protection path-segment for it in the event of failure. The protected span is in a “straddling” relationship to the other shorter p -cycle, which offers two protection path-segments to it. Three working paths, shown in Figure 4.1(b), cross the span from *Zurich* to *Prague* in the normal network state: i.e., *Zurich–Prague*, *Paris–Zurich–Prague* and *Copenhagen–Prague–Zurich–Milan*.

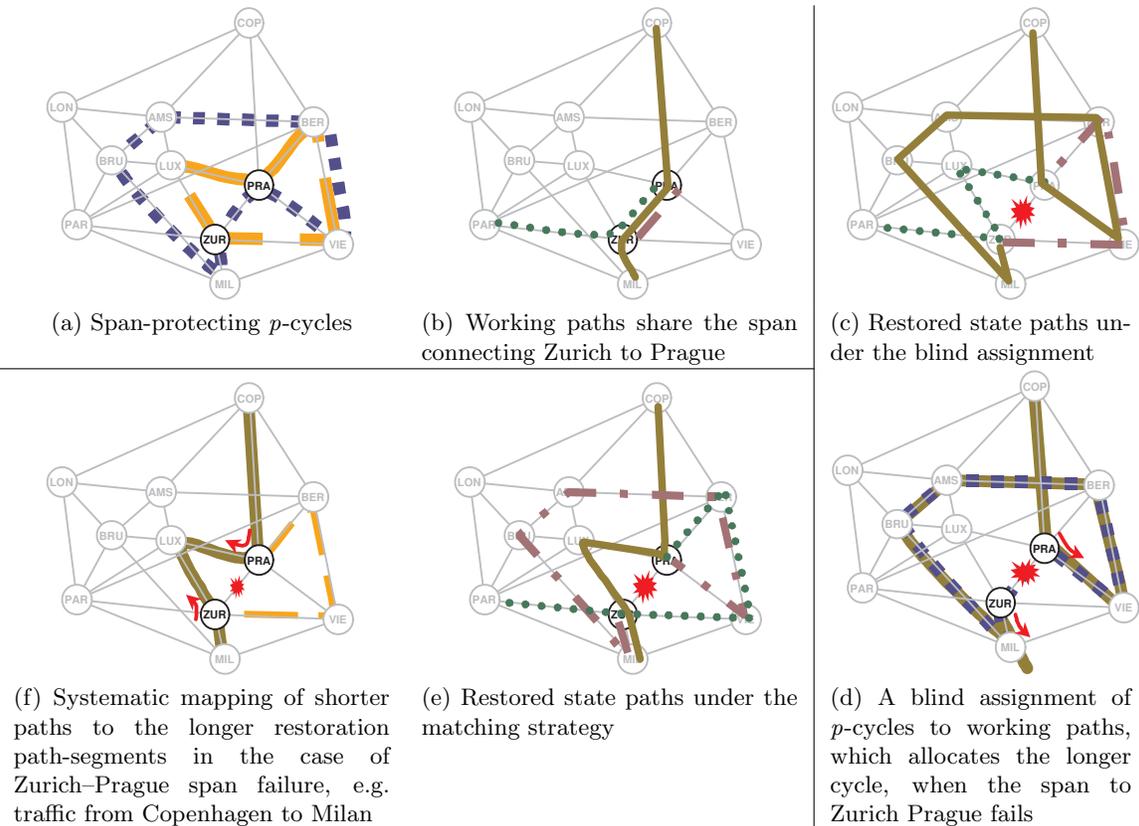


Figure 4.1: Matching Long Working Paths with Short Path-Segments in the Available p -Cycles.

4. Fully Transparent p -Cycle Designs

In total, there are enough protection path-segments to fully protect those working paths against failures of the span in question. But whether all affected working paths wind up with suitably limited end-to-end optical path lengths will depend on the *allocation* of the available protection path-segments. To illustrate the basic concept, Figures 4.1(c) and 4.1(e) portray two different assignments of available p -cycles to working paths. To simplify the illustration, *only* restored states for the path *Copenhagen–Milan* are shown in 4.1(d) and 4.1(f). This restored state path, from *Copenhagen* to *Milan*, is clearly very long in the arbitrary assignment 4.1(c)-(d). The failed channel on the longest working path is associated with the longest protection segment possible through the two available p -cycles shown for the given failure.

Although perhaps not a preferred mapping, there would be many situations where it would not technically matter if this mapping arose. For instance, the BLSR-ring operating principle always imposes—in effect—the worst mapping; i.e. an entire loopback around the ring. But once operational, this does not actually matter as long as the ring size obeys length design limits. In trying to adhere to some maximum transparent domain length limit that is as large as possible, it seems obvious that a better association of working paths to protecting p -cycles is possible. An example is the mapping in Figure 4.1(e)-(f), which greatly reduces the end-to-end length of the restored state path in question. As long as preferred mapping for the path illustrated can be obtained without requiring other affected paths to exceed length limits, then that is the preferable idea pursued.

Matching longer working paths with shorter path-segments through available p -cycles in the event of failure is a simple principle to state. Although simple in concept, one reason this approach has not been developed yet for p -cycles is that the basic design problem for p -cycles (as with other span-protecting principles) is in a framework that is not required to resolve path information, only to protect working channels (counts per span). Technically, there is considerable added complexity to bring the end-to-end routing of every working path into visibility in the design problem. A primary related design problem can be limited to just making the best associations within an existing otherwise ordinary set of minimum-capacity p -cycles. But then, further designs will go on to also *choose* the p -cycles that best suit the overall strategy, ideally with little or no increase in the overall spare capacity.

4.1.3 ILP Models for Complementary Matching of Working Paths and Protection Segments

Table 4.1 numbers and summarizes all ILP design models formulated in this chapter. ILP₁₄₋₁₇ mathematical models are aimed at matching, in various ways, the longer working paths crossing a span with the shorter path-segments through p -cycles protecting the span in question. ILP₁₄ first performs an optimal assignment of protection path-segments to working paths, within an otherwise ordinary set of p -cycles produced by any instance of ILP₁ for conventional minimum spare capacity design, in a way that minimizes the average lengths of all restored path states. ILP₁₅ is a variant that minimizes the length of the longest optical path in the restored network state. ILP₁₆ introduces bi-criterion objectives to optimize the intelligent matching effect while retaining minimal spare capacities and minimized optical path lengths. And ILP₁₇ produces further designs having an explicit maximum length limit on paths in the restored state.

Table 4.1: *Summary of the ILP Mathematical Models Introduced in this Chapter*

Model	Problem Description	ILP Formulation
ILP ₁₄	Minimum average restored state path length model, given a 100% span restorable network design as input.	Assuming an existing set of p -cycles and working paths—equations (4.1)-(4.4).
ILP ₁₅	Minimum longest restored state path.	Equations (4.2)-(4.7).
ILP ₁₆	p -Cycle minimum capacity planning under overall optical length minimization.	Bi-criterion minimization of both capacity and path lengths—equations (4.2)-(4.9).
ILP ₁₇	ILP design model constraining optical paths under a specific fixed limit.	ILP ₁₈ plus absolute length limit—ILP ₁₈ and equations (4.7) and (4.11).
ILP ₁₈	Minimum spare capacity design model for whole fiber switched p -cycles.	Equations (4.14)-(4.20).

4.1.3.1 ILP₁₄ Minimizing the Total Length of Restored Lightpaths in a Conventionally Designed p -Cycle Network

The ILP₁₄ mathematical model is defined for an intelligent assignment of protection path-segments to failure-affected working paths in an otherwise conventional 100% span-restorable p -cycle network design optimized for minimum spare capacity. Thus, the scope here is to change neither the spare capacity, nor p -cycles present, nor the routing of working paths.

4. Fully Transparent p -Cycle Designs

Instead, ILP₁₄ keeps all those “as is” and do the best matching that can be done in any given p -cycle network design, in order to minimize the total (and hence average) length of optical paths over all possible restored state paths of the network. The following defines ILP₁₄ mathematical symbology.

Sets As previously stated, S , D and P represent spans, demand relations and candidate cycles available across the network. Another set $B = \{\text{left}, \text{right}\}$, indexed by b , is required to differentiate the two sides of each p -cycle under span failure conditions. Given a failed span straddling a specific p -cycle, the shorter protection path-segment is considered here as the left side of the p -cycle, while the longer one is considered as its right side. Each on-cycle span has uniquely a left side, i.e., there is no right side.

Input Parameters As previously defined, d^r remain the number of units of capacity for demand relation r and δ_j^r indicate the spans that they cross under normal network conditions. A new parameter $\mathfrak{S}_{i,j}^p \in B$ encodes spans along protection path-segments provided by p -cycle p to a failed span i : $\mathfrak{S}_{i,j}^p = b$ if side b of the p -cycle p crosses span j in the event of span i failure, and $\mathfrak{S}_{i,j}^p \neq b$ otherwise. Thus, $\mathfrak{S}_{i,j}^p$ is always left for on-cycle failed spans, since p -cycle p does not have a right side for the spans crossed en route (i.e. $x_j^p = 1$). Also, the pre-computation of input parameters $\mathfrak{S}_{i,j}^p$ requires information about span end-nodes (i.e. μ_i^k), which is part of the traditional preprocessing network data file.

Decision Variables Beside the number of unit-sized copies η^p for p -cycle p that are effectively used in the design,

- $n_{r,i}^{p,b}$ indicates, when span i fails, the number of restoration path-segments from p -cycle p that are used on its side b for demand relation r .
- $L_{r,i}^{p,b}$ encodes the (end-to-end) optical length of restored state paths for demand relation r , which results from the usage of side b of p -cycle p , in the event of span i failure.
- I^{node} is a constant giving the length-equivalent insertion loss of each node en route to any path. [GGC⁺07] indicates that the optical multiplexer-demultiplexer and transparent cross-connect core may produce an insertion loss per node equivalent to ~ 80 km of fiber length, with typical optically transparent node equipment.

ILP₁₄ Formulation

$$\text{Minimize} \quad \sum_{r \in D, i \in S, p \in P, b \in B : \delta_i^r = 1 \text{ and } \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0} L_{r,i}^{p,b} \cdot n_{r,i}^{p,b}. \quad (4.1)$$

$$\sum_{p \in P, b \in B : \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0} n_{r,i}^{p,b} = d^r, \forall r \in D, \forall i \in S : \delta_i^r = 1. \quad (4.2)$$

$$\eta^p \geq \sum_{r \in D} n_{r,i}^{p,b}, \forall i \in S, \forall p \in P, \forall b \in B : \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0. \quad (4.3)$$

$$L_{r,i}^{p,b} = \sum_{j \in S: j \neq i \text{ and } (\delta_j^r \text{ or } \mathfrak{S}_{i,j}^p = b)} (I^{\text{node}} + C_j) - I^{\text{node}}, \forall r \in D, \forall i \in S, \forall p \in P, \forall b \in B : \quad (4.4)$$

$$\delta_i^r = 1 \text{ and } \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0.$$

Equations (4.1)–(4.4) comprise ILP₁₄. The objective function (4.1) minimizes the total length of end-to-end optical paths in the restored state. The total is run over all affected paths arising from all single span failure states. Equation (4.2) allocates a restoration path-segment to every working path crossing a failed span. Equation (4.3) constrains the number of path-segments handling span failures in a given p -cycle to not exceed its number of unit copies in the design. Equation (4.4) records statistics on the resulting restored state path lengths. In (4.1)–(4.4), the mathematical expression $\sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0$ guarantees the existence of backup b , certifying that protection segments assigned to working routes not go through the right side for on-cycle span failures as an on-cycle span cannot expect more than one protection route over the cycle in question.

4.1.3.2 ILP₁₅ Minimizing the Maximum Length of Restored Lightpaths

The effect of equation (4.1) is to provide a general improvement to the length of paths in the restored state, in order to use less equipment on average. A different viewpoint would be to say that from an optical reach performance standpoint, minimizing the single longest path that arises anywhere in the network for any given restored state would be a more important objective. This gives rise to a “minimax” type of optimization problem, in which equations (4.5)–(4.6) replace the objective function (4.1) and equation (4.7) states that

4. Fully Transparent p -Cycle Designs

the minimax arises out of *used* protection path-segments only, with constraints (4.2)-(4.4) remaining the same.

$$\text{Minimize } X. \quad (4.5)$$

$$X \geq L_{r,i}^{p,b} \cdot \wp_{r,i}^{p,b}, \quad (4.6)$$

$$\forall r \in D, \forall i \in S, \forall p \in P, \forall b \in B: \delta_i^r = 1 \text{ and } \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0.$$

$$n_{r,i}^{p,b} \leq \wp_{r,i}^{p,b} \cdot M^\infty, \forall r \in D, \forall i \in S, \forall p \in P, \forall b \in B: \delta_i^r = 1 \text{ and } \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0. \quad (4.7)$$

Additional Input Parameters The following definitions complete ILP₁₅ formulation.

- $\wp_{r,i}^{p,b} \in \{0, 1\}$ records the usage of restoration path-segments in the models that seek the least maximum optical path length under minimum capacity costs or constrains the path under fixed maximums: $\wp_{r,i}^{p,b} = 1$ when at least one unit-sized copy of side b of p -cycle p is committed for restoring traffic from demand relation r , in the failure of span i (i.e. $\wp_{r,i}^{p,b} = 1$ if $n_{r,i}^{p,b} > 0$); and $\wp_{r,i}^{p,b} = 0$ otherwise.
- M^∞ is a suitable large constant that serves as a surrogate for infinity.

4.1.3.3 ILP₁₆ for Bi-criteria Optimization of Capacity and Restored State Paths

ILP₁₄ optimizes the matching of restoration path-segments with working paths in any given or existing p -cycle protected network. Knowing that the p -cycle minimum capacity design model might have many different solutions for the same capacity costs, one can obtain a refined model that will select a solution of equal or similar cost to the optimal capacity solution, but which potentially can further minimize the restored state path length. This involves combining ILP₁ and 14. The result, ILP₁₆, keeps constraints (4.2)-(4.4) as above, but introduces the bi-criterion objective function (4.8) with accommodations in (4.9). The aim is still to minimize spare capacity as the primarily goal, but now to do so with the concern for reducing optical path lengths as well. Doing so, equation (4.3) now calculates

4. Fully Transparent p -Cycle Designs

the number of unit-sized copies of selected p -cycles, and constrains as well the number of protection path-segments assigned under what is available as in ILP₁₄.

In one mode of use, the parameter α is chosen to be small enough not to upset the principal objective of staying at minimal spare capacity, while biasing the design towards the selection of cycles that will reduce lengths of restored state paths; we mainly use the model in this mode. The other mode is where α is made large enough, and varied, to produce the set of Pareto-optimal trade-off points between ever lower average path lengths and increasing total spare capacity. As usual, C_j indicates the cost of one unit-channel on span j and s_j encodes the number of spare capacity channels placed on j in the design.

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j + \alpha \cdot \sum_{r \in D, i \in S, p \in P, b \in B : \delta_i^r = 1 \text{ and } \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0} L_{r,i}^{p,b} \cdot n_{r,i}^{p,b} \quad (4.8)$$

$$s_j = \sum_{p \in P: x_j^p = 1} \eta^p, \quad \forall j \in S. \quad (4.9)$$

Similar improvements can be obtained by merging the ILP₁₅ design model, instead of ILP₁₄ model, with the conventional ILP₁. The bi-criterion objective (4.10) will be aimed at limiting the maximum length of restoration paths while also minimizing the total required spare capacity.

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j + \alpha \cdot X \quad (4.10)$$

4.1.3.4 ILP₁₇ Asserting an Absolute Transparent Path Length Limit

Equations (4.7) and (4.11) can also be added to strictly disallow paths over some fixed length limit “ L_{\max} ”. Note that this also freely allows any path length up to the limit, whereas minimax forces all paths to be lower than the lowest possible longest path length. One can either use the bi-criterion objective function (4.8) or focus on minimizing spare capacity only [equation (2.1)]. The actual difference between this model and ILP₁₅ is that here, the lengths of all optical lightpaths are constrained under the absolute limit L_{\max} and one allows that doing so may involve an increase of the total spare capacity. This is in contrast to ILP₁₅, where transparent distance was minimized without extra capacity over minimum cost requirements.

$$L_{r,i}^{p,b} \leq L_{\max} \cdot \varphi_{r,i}^{p,b}, \forall r \in D, \forall i \in S, \forall p \in P, \forall b \in B: \delta_i^r = 1 \text{ and } \sum_{j \in S: j \neq i \text{ and } \mathfrak{S}_{i,j}^p = b} 1 > 0. \quad (4.11)$$

In asserting absolute path length limits, one might want to know the shortest optical reach limit for which a design solution is feasible. The bi-criterion objective is reversed in order to shorten the longest path as much as possible in design before minimizing spare capacity requirements.

$$\text{Minimize } X + \alpha \cdot \sum_{j \in S} C_j \cdot s_j. \quad (4.12)$$

4.1.4 Case Studies and Experimental Results

In the series of experiments conducted to assess the performance of the novel matching strategy, the conventional p -cycle minimum spare capacity design model, i.e. ILP₁, was first solved for the well-known pan-European **Cost239** network instance. Following the completion of ILP₁, several matching policies were applied to the obtained p -cycle design solution. The new policy associated longer working paths with shorter protection path-segments through available p -cycles, using in turn ILP₁₄ that minimizes the average length of restored state paths and ILP₁₅ that minimizes the longest (instead of “average”) restored state path length in the design. Two more policies were tested for comparison purposes; one *randomly* assigned protection path-segments to working paths and the other associated *long protection segments with long working paths*. Further sets of experiments then considered length optimization aspects for direct control optical path lengths in the restored network state. More specifically, ILP₁₆ biased the reference p -cycle minimum spare capacity problem towards selecting p -cycles that also minimize the average or the longest restored state path length, and ILP₁₇ was used to assert optical reach limits in the design.

Table 4.2 reports capacity requirements in the design, the ratio of total distance-weighted spare capacity to total working, statistics on minimum, average and maximum restored state path lengths, as well as the percentage of optical paths that are longer than 3000 km, a typical maximum transmission distance (MTD) for transparent optical network exercises [GLW⁺06]. The second column is for working paths; the third set of columns pertains to the new matching strategy under average length optimization: the first data set

4. Fully Transparent p -Cycle Designs

assumes a pre-generated conventional design and the second data set corresponds to the bi-criterion approach. The fourth column is similar to the first data set in column 3, but ILP₁₄ is substituted for ILP₁₅ in order to shorten the longest restored state path in the design. The fifth column records results for a 4000 km reach limit, and the sixth set of columns relates to comparative matching policies.

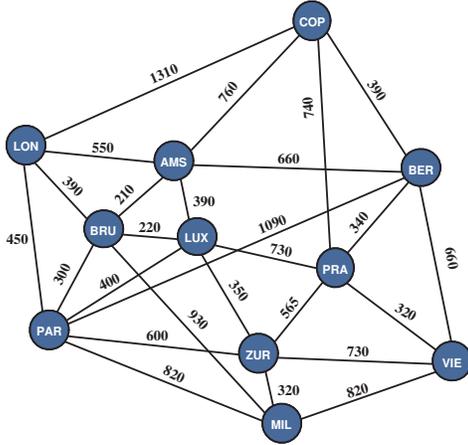
Table 4.2: *Sample Results for Restored State Path Length Minimization in the Cost239*

Metrics	Normal State Paths	Average Length Optimization		Minimax (ILP ₁₅)	ILP ₁₇ asserting a 4000 km reach limit	Other Assignments	
		ILP ₁₄	ILP ₁₆			random	long paths—long cycles
Capacity	working	spare					
channel-kms	137,170 (n/a)	85,640 (62%)	85,640 (62%)	85,640 (62%)	101,725 (74%)	85,640 (62%)	85,640 (62%)
Length (km)	Optical transmission length of non-failed and restored state paths						
- Min.	210	510	510	700	770	770	680
- Avg.	833	3825	3786	4148	2724	4213	4539
- Max.	1900	7955	7625	6665	3990	7555	7815
-Paths longer than 3000 km	n/a	180 (61%)	190 (64%)	216 (73%)	139 (47%)	223 (76%)	238 (81%)

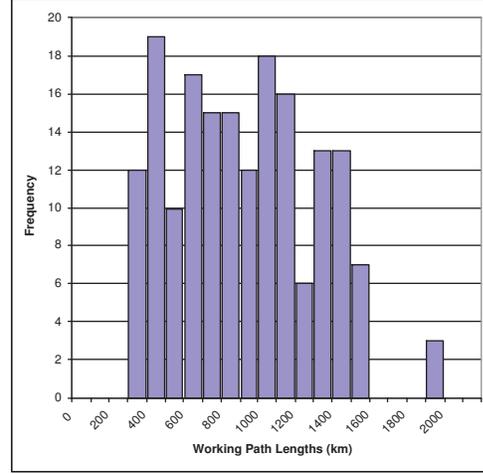
4.1.4.1 Characterization of (Non-Failed) Working Paths

Figure 4.2 gives a mean to data for working path lengths, in the second column of Table 4.2—4.2(a) specifies span lengths across the Cost239 network and histogram 4.2(b) records statistics for the optical length of working lightpaths following an initial shortest distance weighted routing. The length of an optical path is the sum of the physical length (in kms) of spans crossed en route *plus* an additional 80 km insertion loss estimate used for every node equipment transited. Within the ILP symbology defined for this work, the optical length of a path is computed as in equation (4.13). Statistically, the resulting non-failed shortest distance paths are in between 210 km and 1900 km, for an average normal state length of 833 km. The incurred total working capacity was 137,170 channel-kms.

$$L^r = \sum_{j \in S: \delta_j^r = 1} (I^{node} + C_j), \forall r \in D. \quad (4.13)$$



(a) Span Lengths



(b) Characterization of (non-failed) working paths, following a shortest distance weighed routing

Figure 4.2: Optical Transmission Length of Working Paths (alone) in the COST239 Network.

4.1.4.2 Minimizing Average versus Minimax on the Length of LightPaths

Table 4.2 shows that ILP₁₄ provides a general improvement of restoration lightpaths, with average length of 3825 km and 61% paths over 3000 km, taken as reference criterion for what could be considered excessively long paths. ILP₁₅ achieves a higher average path length (4148 km) and 73% of restored paths over 3000 km; this is because many paths tend to be increased in length to accomodate the shortest possible *longest* length path, and statistically squeezed-up to just below that maximum value. In comparison, ILP₁₄ that minimizes the average path length in the restored network state result in a few protection paths longer than (*say*) 7000 km; but considered all together many fewer paths tend to be as long individually as with ILP₁₅. Recall, however, that so far both ILP₁₄ and 15 are solved only within the existing design from ILP₁.

Henceforth, we consider only model variants that minimize the average path length in the restored state. This will restrict the scope of subsequent considerations of both cases where the context involves trying to manage restored path lengths in an already existing set of working paths and p -cycles or models where one has full design control. Some

justification for being more interested in the minimum average length case, as opposed to the minimax, is given in Section 4.1.5. But clearly the methods that are developed here could similarly be worked through for the minimax case, should the reader wish.

4.1.4.3 Effectiveness of Matching Long Working Paths with Short Protection Segments within a Conventionally Designed p -Cycle Network

In one series of experiments, ILP₁ is solved first and ILP₁₄ is adapted in various ways testing three different policies associating working paths with protection path-segments available only from pre-existing solution cycles. Corresponding results are in the third (first data set) and sixth sets of columns of Table 4.2. Accordingly, matching shorter protection segments with longer working paths does provide the best average length and the fewest number of restored state paths of “long” length, again using 3000 km as a criterion; the maximum path length (not surprisingly) increases slightly but inconsequentially. So the matching of long working paths with short protection segments is of some benefit, even just as a way of deciding which protection path-segments from which p -cycles to use for each failure in a pre-existing design solution. But when working within the confines of an already existing set of p -cycles, the effectiveness of path length reduction using the new matching principle is somewhat limited.

4.1.4.4 Optical Length Optimization using a Bi-criterion Approach

The ILP₁₆ design model that minimizes the total length of optical paths in bi-criterion weighting against capacity cost used $\alpha = 10^{-6}$, which was found to be suitable to control path length as a combined objective without upsetting the value of minimum spare capacity. The second data set in the third column of Table 4.2 summarizes statistics on the resultant path lengths in this case. The average path length is slightly lower than in a comparative instance with ILP₁₄, and the slight reduction in average (restored state) path length is obtained through some significant reduction in the shortest path(s) and a slight reduction in the longest path(s) as well. But the overall effect in terms of path shortening is not great. The result tends to suggest that minimizing average path length at minimum spare capacity with the detailed model in ILP₁₆ may not yield much improvement over simply designing for minimum capacity and adjusting path lengths within those structures. This could be

true in general if path lengths are simply strongly dictated or dominated by properties of any minimum capacity design instance.

However, there can be many different equivalent minimum spare capacity designs but only one (essentially at random) is considered for the corresponding ILP₁₄ design result in Table 4.2. There could be many other ILP₁₄ outcomes, starting with different initial minimum-capacity designs that do not do as well as the ILP₁₆ model does here. As a check on this, one extra ILP₁ solution (which is also at minimum capacity) followed by the application of ILP₁₄ was found with the result being an average of 3935 km and a significantly greater maximum path length of 8035 km. A benefit of ILP₁₆ is that it assuredly produces a solution with average path length statistics, which are the best possible under minimum spare capacity, whereas any given succession of an ILP₁ solution followed by an ILP₁₄ accommodation does not have this guarantee. Additionally, ILP₁₆ can also be used with high α values, which allow entering the domain of trading minimal addition spare capacity for further reduced path length average statistics, and this is not an option with a combination of the ILP₁ and 14 models.

4.1.4.5 Network Design with Direct Optical Path Length Control

Perhaps the most direct way to control the path length is to simply stipulate a tolerable maximum using ILP₁₇. With it we have a means to simply assert a maximum optical path length limit in the restored state and still achieve absolute minimum spare capacity, if that remains technically feasible for the given length limit. Also, ILP₁₇ is allowed to increase the spare capacity requirements over the absolute minimum spare capacity depending on the optical reach limit asserted. For instance, the fifth column of Table 4.2 shows that constraining optical path lengths under 4000 km does increase capacity redundancy to 74%. This is ~19% extra spare capacity over what is required in the strictly minimum spare capacity designs made with ILP₁₈ and 20 models. It is interesting to note, however, that with the addition of this ~19% extra spare capacity, every working and restored state path is under 4000 km. As a design tool this is perhaps the most general methodology for how to design a completely protected transparent island using ULH technology.

Figure 4.3 further portrays the generalized trade-off of spare capacity for optical path length control, when trading capacity to achieve any given limit on maximum optical reach.

ILP₁₇ was solved repeatedly for several fixed length limits and the resulting excess capacity to satisfy the design goal is plotted. As expected, the absolute minimum spare capacity is achievable with suitably large optical reach limits and the problem becomes infeasible when trying to assert very low reach limits. The minimal tolerable path length limit at which ILP₁₇ is feasible was actually found to be 1900 km. The shortest reach limit at which strictly minimal spare capacity can still be realized is 6750 km. Comparing this with maximum optical path lengths for ILP_{14–16} (see Table 4.2), it seems that allowing ILP₁₇ to change its selection of p -cycles to accommodate fixed path length limits can achieve the same redundancy as minimum cost design, but with a lower maximum path length.

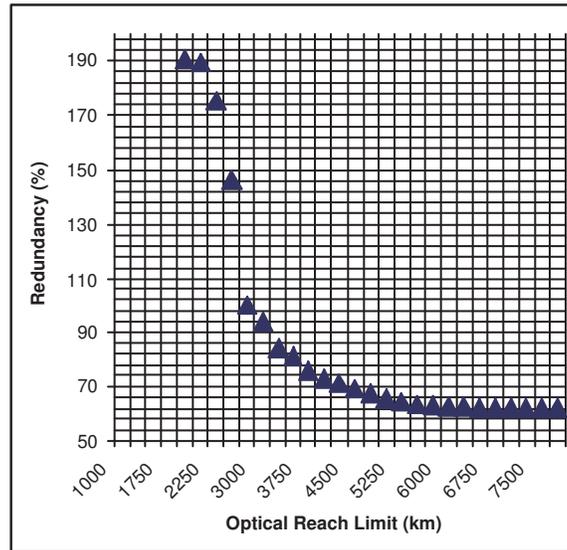


Figure 4.3: Effects of Asserting a Direct Optical Reach Limit on Spare Capacity Requirements

4.1.5 Which is Preferable—Lower Average Path Length, Shortest Minimax or Fixed Optical Reach Limit?

Given the variations that are possible with the idea of systematically matching longer working paths with shorter protection path-segments (found in p -cycles chosen for a fully transparent design), the following question remains: *is it preferable to have a lower average path length or the shortest minimax path length, or to simply have all paths below*

a set limit even if this requires extra capacity? In practice, the choice would depend on the network circumstances— If many of the path lengths were already over the distance requiring regeneration, then minimizing the average path length corresponds to the least total consumption of channel-kms and regenerator costs. If one was already close to having every path below a transparent limit to start with in the restored state, then squeezing the longest path to its minimum could enable an entirely transparent solution (although this is a solution in which numerous other paths are individually longer than they would have to have been otherwise). But the work also showed that it is possible to just assert an upper length limit on all paths based on technological capabilities, and solve for the p -cycles and path assignments and/or added capacity to satisfy that requirement. This last option seems to be the most general and straightforward engineering approach. Although it may require some extra spare capacity, it would be the most direct method to design an entire transparent island based on a given ULH technology capability, such as, say, a 4000 km reach. In this case, an entire transparent island can be defined in order to fully route and protect all demands inside itself in a fully transparent manner.

4.2 p -Cycle Protection at the Glass Fiber Level

The p -cycle literature typically assumes a protection switching based on the wavelength granularity level, which is more than attractive from the perspectives of flexibility in the routing of working paths and freedom in the selection of cycle structures. [SSG03a] distinguishes between the three types of configuration in Figure 4.4—case 4.4(a) assumes full *opaque* network conditions; wavelength discontinuities imply o-e-o conversion capabilities at every node across working paths and along p -cycles, meaning that a given working path or p -cycle leaves and re-enters the optical domain to access the next span on which it probably rides onto a different wavelength. In the configuration type 4.4(b) referred to as *hybrid*, pre-failure paths and p -cycles are independently transparent and consequently use the same wavelengths from end-to-end. But a given p -cycle is not required to ride onto the same wavelength as the working paths it handles. So wavelength conversion is partially required and located at the entry points of failed paths into p -cycles. As with hybrids, working paths and p -cycles are transparent in 4.4(c); but the same wavelength used by any given working path is now also required for its protecting cycles. Post-failure states only assume

an optical switching and the bypass of electrical switches (i.e. no wavelength conversion in the electronic domain). To be effective, such a *fully transparent p -cycle* design requires the use of two different types of fibers: one dedicated to normal state routing (only) and the other to protection channels. (Please refer to 2.25 for pre-failure illustrations.)

But wavelength switching operations greatly increase equipment prices and involve more complexity in the design exercise. The preliminary study [GGC⁺09] brought to attention that where wavelength conversion is allowed, either at p -cycle entry points or at every node crossed en route, high equipment costs are incurred due to o-e-o conversion from one span to the next or from surviving path-portions to p -cycles (and vice versa). And nevertheless fully transparent p -cycle designs eliminate the need for electrical switching, the cost of network nodes will increase because of the proliferation of working and protection fiber optics, as working paths and p -cycles no longer ride onto the same fibers. Furthermore, wavelength assignment and continuity requirements also bring computational issues and much more complexity in fully transparent p -cycle designs.

4.2.1 Chronological History of Glass Switched p -Cycles and Objectives

Besides transparent reach concerns, the cost and complexity of wavelength assignment and conversion and wavelength-selective switching are always of primary consideration in the design of survivable networks. As mentioned in [Gro03b], p -cycle structures can be configured on a waveband basis, as opposed to wavelength, with each waveband of wavelengths treated as a single unit. The specific case where whole fiber optics define wavebands is referred to as *whole fiber* (or simply, “*glass*”) *switched p -cycles*. p -Cycles at a fiber level of protection are exciting in that to protect against fiber failures or span cuts, wavelength assignment within the failed fibers is irrelevant as long as p -cycle fibers support the same waveband. This means despite the general recognition that requiring wavelength continuity greatly complicates the basic service routing problem, there is no further complication due to protection considerations if p -cycles are used at the fiber-switching level to protect fully transparent transport networks. Implicitly, every wavelength assignment actually retains continuity under protection rerouting because the wavelength in question is by definition free for use on the fiber dedicated for protection (if not already protecting another failure).

4. Fully Transparent p -Cycle Designs

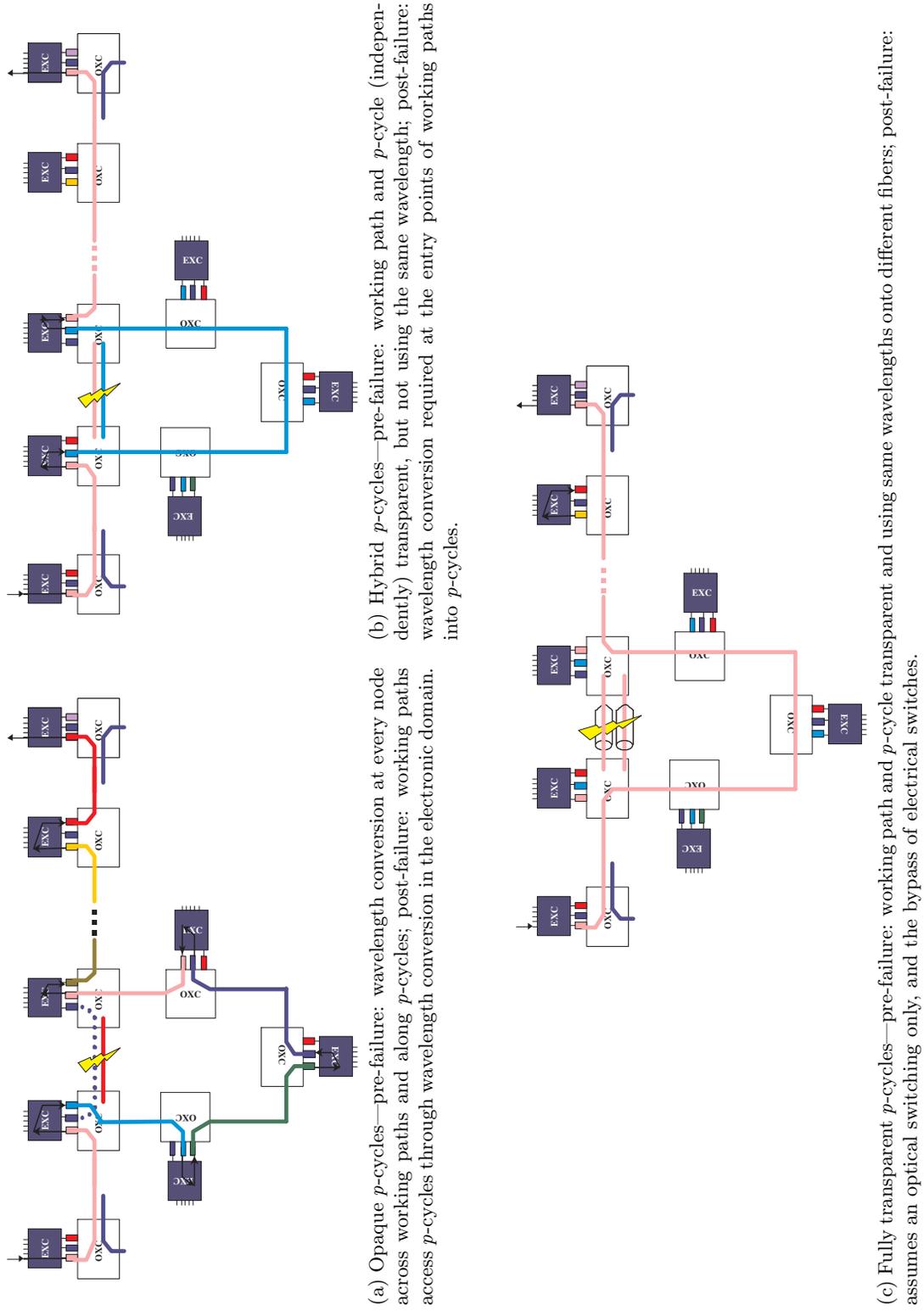


Figure 4.4: p -Cycle Configuration Types in the WDM Layer.

If [Gro03b] opened the possibilities for fiber-level protection with p -cycles, the seeds of the idea have remained unexplored. Perhaps the main reason why glass switched p -cycles have not been seriously challenged is a widespread idea that whole fiber switching operations are very slow. This study proposes an actual state-of-the-art whole fiber cross-connect switch technology. Several commercial examples of whole fiber cross-connect switches will be reviewed with the purpose of supporting the practical feasibility of glass switched p -cycles. We will also venture a comparison of whole fiber cross connect switches with OXCs from the perspective of costs incurred, with an estimate of 10% for the cost of ports on the hypothetical fiber switch over the cost of wavelength selective ports on a traditional OXC.

Although unintentional, [SG04a] generated the first whole fiber switched p -cycle designs as p -cycle solutions for homogeneous networks, of which spans consist of exactly two fiber optics with identical number of wavelength channels. So only Hamiltonian cycle structures were involved in the solutions; even though an effective p -cycle network design can be based on a single Hamiltonian, with the attraction of a quite easy calculation, the preliminary study [OG07] previously demonstrated that designs involving many complementary cycle structures require much less spare capacity. Another objective of this research is to revise the basic p -cycle design problem in a way matching fiber-level protection paradigms. But rather than using only Hamiltonian cycle structures, as [SG04a] did, all possible candidate cycles will be now considered as eligible. The complexity of resulting mathematical formulations will also be compared to that of wavelength switched p -cycles.

4.2.2 Whole Fiber Cross-Connect Switches

The idea behind glass switched p -cycles is to keep a flexible normal state routing as with wavelength switched p -cycles but use, in place of OXCs, cross-connect devices that have the ability to switch (at once) all wavelengths of an entire failed fiber optic into a p -cycle formed out of span fibers. In the past, glass-switching fabrics used to be very slow in practice (1s or more), bulky, expensive and typically limited to 16x16 or 2x48 matrix sizes. Recent developments in the fiber optic industry gave rise to a new generation of whole fiber switches, which offer the intended capability (i.e. the ability to switch optical signals transparently and independently of data rates, formats, wavelengths, protocols and services) with faster switching times, more compact form-factors and larger matrix sizes.

4.2.2.1 Actual State-of-the-art

Technological solutions for fiber cross-connect switches relate to micro-electro-mechanical system (MEMS) optical switches, which utilize micro-mirrors to switch or reflect a given optical signal from one fiber to another depending on the relative angle of the micro-mirror. MEMS allow to switch many optical channels in a relatively small amount of space; actual commercial examples of glass switches in [Glia] use MEMS micro-mirrors to transparently interconnect matrices of 24x24 to 192x192 single mode fibers. All SONET/SDH, Ethernet, digital and analog signal formats are supported, with traffic data rates reaching up to OC-768, 10GE and DWDM. References [CZKC, DSCB04, FSO⁺] provide more details on technological aspects for the design, fabrication, testing and manufacturing of the 64x64 and 80x80 3D-MEMS switch fabrics.

[DSCB04] also addresses the question of MEMS optical switch control system speeds. It reports switching times tightly distributed around the mean value 9 ms, with a peak of 20 ms, for 64x64 3D-MEMS. Such response times are found consistent with the duration of resonant frequencies of 300, 375 and 450 Hz used in the 64x64 3D-MEMS control system. Whole fiber cross-connect devices in [Glib] corroborate the 20 ms (or less) switching time expectations, with 2.2 dB insertion loss and high optical return loss, for matrix sizes of 24 and 48 whole single mode fibers (1270 to 1630 nm). Glass fiber cross-connect devices in [Glib] also include input and output power monitoring which could be used to activate p -cycle switching actions. The idea of glass switched p -cycles can be pursued on the basis of available whole fiber cross-connects, with fast switching times and low insertion losses.

4.2.2.2 Hypothetical Cost Reduction

Because the technology for switching whole fibers is much older and “*lower-tech*” than that for DWDM switching, it is reasonable to expect that small glass-switching cross-connects, or even simply three-port glass switching devices serving as p -cycle nodes, would be very much lower in cost than the corresponding number of DWDM channel ports on a DWDM wavelength path switching cross-connect. Hypothetically, say that a fiber is carrying even just 10 wavelength channels that have to all be rerouted following a failure. To replace the glass fiber (in each direction) requires two fiber interface ports and one glass-switch crosspoint. No optical filters, demultiplexers or combiners are needed. To do the

same function at the DWDM layer requires (with a transparent optical switch) two fiber interface ports, optical demultiplexer, 10 DWDM qualified optical switchpoints, optical remultiplexing, and wavelength selective optical filters as the technology requires. Because the one whole-fiber switch is a simpler technology than each of the ten DWDM crosspoints, *the whole fiber switch stands the prospects of being $\sim 10x$ less costly than the DWDM switch for the same function* even if we overlook the optical mux/demux requirements of the DWDM switch. Both options require remote telemetry and control, but again the glass fiber switch requires one command rather than ten. The DWDM layer may be waveband switched, which alters the argument, but on the other hand it is restored again if the fiber is bearing not 10 but 100 channels in ten wavebands, and so on.

4.2.3 Complexity Reduction of Fully Transparent p -Cycle Design Models

Sets In addition to the sets of spans S , demands D and candidate cycles P , the set C encodes wavelengths available into a fiber and is indexed by l . All fiber optics across a given network are assumed the same waveband of $|C|$ wavelengths, where $|C|$ is the cardinality of set C .

Input Parameters As with prior ILP-symbology, C_j is the cost of each channel (i.e. capacity unit) on span j , but will also refer to the length of span j in the capital expenditure (CapEx) context; w_i is the number of working channels to be protected on span i following whatever routing of the demands; d^r is the number of units of capacity for demand relation r ; $\delta_j^r \in \{0, 1\}$ indicates spans that each given demand r crosses en route under no-failure states (i.e. $\delta_j^r = 1$ if r crosses j and $\delta_j^r = 0$ otherwise); and $x_i^p \in \{0, 1, 2\}$ encodes the number of protection segments that one unit-sized copy of p -cycle p provides to span i (i.e. $x_i^p = 2$ if i straddles p , $x_i^p = 1$ if i is on p and $x_i^p = 0$ otherwise).

- A new parameter F_{w_j} encodes the number of working fibers which comprise span j .

Decision Variables Also, η^p still represents the number of unit-channel copies of candidate cycle p in the design. In addition,

- F_{s_j} encodes the number of protection fibers which comprise span j in the design.

- $\lambda_l^r \in \{0, 1\}$ records wavelengths assigned to working routes for demand relation r ; $\lambda_l^r = 1$ if r uses wavelength l under normal network states and $\lambda_l^r = 0$ otherwise— λ_l^r is assumed binary to simplify ILP formulation, but this condition imposes the number of units for a given demand-pair to be less than or equal to wavelengths available in each fiber optic.

4.2.3.1 ILP₁₈ for Glass Switched p -Cycle Designs

Equations (4.16)–(4.20) comprise the ILP₁₈ mathematical design model for glass switched p -cycles. The objective function (4.14) minimizes total (working and protection) fiber-kms required in the design. Equation (4.15) assumes an initial routing of demands and derives the number of working fibers to be protected on spans. Constraints (4.16) assigns a single wavelength to every working path, (4.17) ensures that every wavelength is uniquely assigned into each given working fiber and (4.18) keeps the number of wavelengths assigned under what is available within each fiber. Equations (4.19) ensures full span failure restorability and (4.20) places protection fibers on edges.

$$\text{Minimize } \sum_{j \in S} (F_{w_j} + F_{s_j}) \cdot C_j. \quad (4.14)$$

$$F_{w_j} \geq \frac{w_j}{|C|}, \forall j \in S. \quad (4.15)$$

$$\text{subject to: } \sum_{l \in C} \lambda_l^r = d^r, \forall r \in D. \quad (4.16)$$

$$\sum_{r \in D} \lambda_l^r \cdot \delta_j^r \leq F_{w_j}, \forall j \in S, \forall l \in C. \quad (4.17)$$

$$\sum_{r \in D, l \in C} \lambda_l^r \cdot \delta_j^r \leq F_{w_j} \cdot |C|, \forall j \in S. \quad (4.18)$$

$$F_{w_i} \leq \sum_{p \in P} (x_i^p \cdot \eta^p), \forall i \in S. \quad (4.19)$$

$$F_{s_j} = \sum_{p \in P: x_j^p=1} \eta^p, \forall j \in S. \quad (4.20)$$

ILP₁₈ appears as an obvious adaptation of the conventional p -cycle ILP₁, where working and protection wavelength variables w_j and s_j are substituted for F_{w_j} and F_{s_j} in order to match fiber-level protection paradigms. Whole fiber protection eliminates wavelength continuity requirements in the restored network state but still requires us to keep the same

4. Fully Transparent p -Cycle Designs

wavelengths across original working paths. Equations (4.16), (4.17) and (4.18) allocate wavelengths to working paths, and make sure that each specific wavelength is uniquely assigned into any fiber optic while not exceeding the number of fibers on each given span.

4.2.3.2 Computational Complexity

One way to measure the computational complexity of an ILP mathematical model is to assess the number of variables and constraints manipulated in specific ILP-related instances. Table 4.3 records variables and constraints for **Havana**, **Cost239**, **Italy**, **Bellcore** and **Euro** test cases, under the assumption of 40 wavelengths per fiber. The last column is for ILP_{18} and the second set of columns relate to opaque, hybrid and fully transparent wavelength switched p -cycle design models given in appendix B. Opaque ILP instances involve less variables and constraints because all other configurations provide any form of signal transparency, implying many more additional variables aimed at accomodating wavelength assignment and wavelength continuity requirements. But ILP_{18} for glass switched p -cycles also involves fewer variables and constraints (even less than half in **Cost239**, **Italy** and **Bellcore** cases) than hybrid and fully transparent wavelength switched p -cycles. The reason is that ILP_{18} imposes wavelength assignment and continuity to working paths, and the same wavelengths are implicitly used under restored network states; this stands in contrast to other forms of transparent designs which require wavelength continuity and assignment constraints for both normal and protection network conditions.

Table 4.3: *Variables and Constraints Involved in Wavelength and Glass Switched p -Cycle ILPs*

Test Case Networks	Wavelength Switched p -Cycles			Glass Switched p -Cycles- ILP_{18}
	Opaque	Hybrid	Transparent	
Havana				
(i) original	3,749 and 79	13,035 and 1,338	13,061 and 2,430	7635 and 1,203
(ii) more traffic	3,749 and 79	18,261 and 1,416	18,287 and 2,508	12,861 and 1,281
Cost239	95,441 and 79	240,366 and 4,731	240,392 and 5,823	99,126 and 1,200
Italy	14,021 and 73	41,371 and 1,692	41,395 and 2,700	19,091 and 1,135
Bellcore	28,416 and 85	74,632 and 2,313	74,660 and 3,489	35,592 and 1,337
Euro	30,225 and 127	84,994 and 2,871	85,036 and 4,635	57,034 and 2,172

4.2.4 CapEx Cost Enhancements

Most network designers directly define cost metrics as capacity requirements in terms of total channel-hops or channel-kms. *Does capacity alone correlate real-world CapEx costs?* A consistent WDM cost model, comprising all relevant pieces of equipment, is crucial to address such a techno-economic question. However, detailed equipment cost values used to be very hard to derive because of such reasons as confidentiality of actual commercial costs, differentiation of product families among vendors, and learning curves affecting equipment year by year. Several partners in the European research project NOBEL (Next generation of Optical network for Broadband European Leadership [Nob]) recently contributed to and agreed on a normalized cost model for WDM pieces of equipment [GLW⁺06, HGMS08].

4.2.4.1 A CapEx Cost Model for WDM Layers

The NOBEL cost model in [GLW⁺06, HGMS08] describes a set of normalized CapEx costs for various network elements and node architectures, including all elements required to build a WDM transport layer. The following derives NOBEL costs for opaque, hybrid and (wavelength or glass switched) fully transparent networks using span-protecting p -cycles.

a. Protection Switching Issues

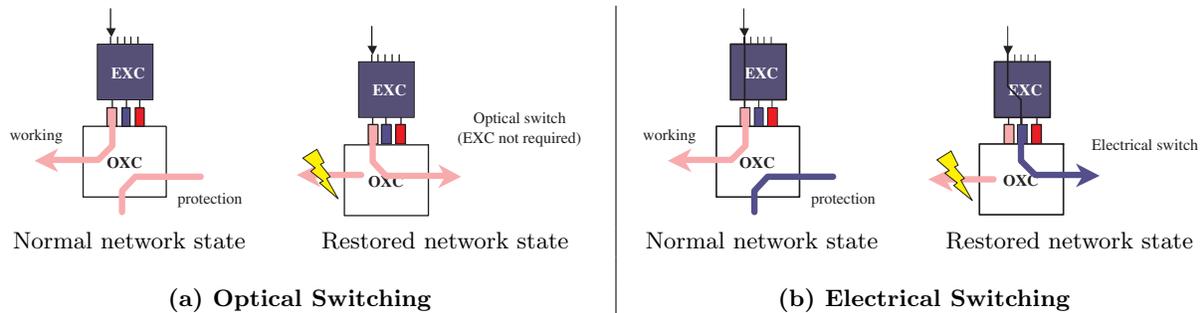


Figure 4.5: *Protection Switching Adaptations of the NOBEL Cost Model, adapted from [Nob].*

A survivable network enables the switching of optical paths in response to failures. Two types of protection switching are typically considered: optical switching and electrical switching. As shown in Figure 4.5(a), optical protection switching utilizes an OXC to switch

4. Fully Transparent p -Cycle Designs

an outgoing signal between two different end-to-end concatenated wavelength paths using a single transponder. This implies that working and protection paths are both transmitted over the same wavelength of a single tunable transponder card. Optically switched paths completely bypass the electrical cross-connect (EXC) and use one single transponder *plus* the standard node switching, when considering (only) one end-node of the protected path.

Tuning the transponder card to a new wavelength using an optical switching is not taken into account because of response time considerations; an electrical switching is used instead. In Figure 4.5(b), an EXC switches the signal in the electrical domain from one transponder to another, meaning that working and protection paths are allowed to use different wavelengths. Enabling electrical switching at one end-node of the protected path requires the use of one working transponder, one protection transponder and three ports on the EXC (one client-side and one for each of the two transponders).

Here are some practical implications of protection switching. Fully transparent span-protecting p -cycles only require an optical switching at the ingress and egress nodes: i.e. two transponder line cards for add/drop operations, one at the ingress and the other at the egress OXC of any lightpath. This stands whether considering a wavelength or a glass switching granularity level, just replacing the OXC by the whole fiber cross-connect device (in case). In contrast, hybrid span-protecting p -cycle configurations require an electrical switching at the entry points of working paths into p -cycles. This implies three EXC ports and two transponders at each of the ingress and egress nodes of any lightpath; one EXC port and one OXC transponder at every intermediate node along working paths, in order to give to span end-nodes the capability of switching onto the protecting structure in the event of failures; one EXC port and one OXC transponder at the entry nodes of every p -cycle. In opaque p -cycles where an electrical switching is required at every node across working paths and p -cycles, corresponding requirements are of three EXC ports and two transponders at each of the ingress and egress nodes for every lightpath; two EXC ports and two transponders at every intermediate node along any working path; two EXC ports and two transponders at every node crossed by any given p -cycle. Opaque and hybrid p -cycle designs can add up to a large difference in EXC ports and transponders, as networks contain an aggregate of hundreds of protected wavelength channels/paths.

b. Equipment Pricing under the NOBEL Cost Model

Figure 4.6 presents a consistent transparent optical path under the NOBEL framework. Basic components comprise: transparent cross-connect devices at the ingress, egress and intermediate nodes; a transparent node amplifier at the entrance/exit of each transparent node; and between two transparent nodes, an inline amplifier (IA), a dispersion compensating fiber (DCF) every 80 km and a dynamic gain equalizer (DGE) every fourth IA.

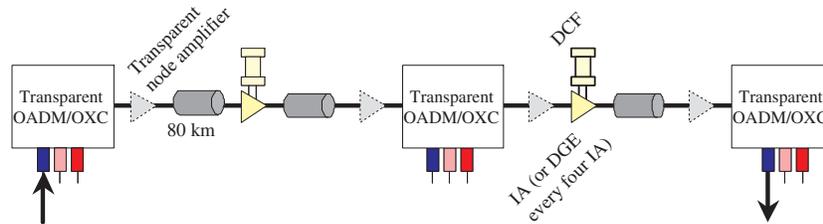


Figure 4.6: Example of Transparent Optical Path under the NOBEL Cost Model for WDM Layers, adapted from [Nob].

Node Base line Architecture

The hybrid transparent node in Figure 4.7 matches protection switching requirements. This is an OXC coupled to an EXC —the latter is only considered for protection switching, meaning that no other functionality such as grooming of lower rate traffic is taken into account in this study. An EXC switching port costs 0.28 under the prevalent data rate of 10 Gbit/s.

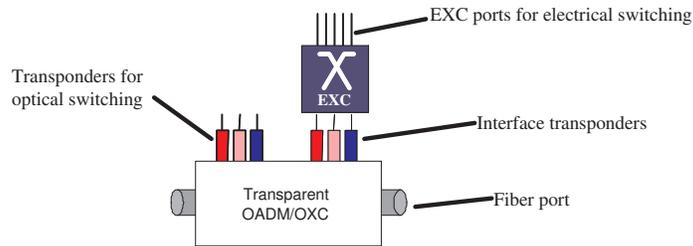


Figure 4.7: A Baseline Node Architecture under the NOBEL Cost Model for WDM Layers, adapted from [Nob].

4. Fully Transparent p -Cycle Designs

Each OXC is priced on the basis of wavebands supported by its incident fiber optics. Under the assumption of 40 wavelengths per fiber, the NOBEL worksheet indicates costs of 11.8 for OADM's (i.e., OXC's of two incident fibers); equations (4.21) and (4.22) calculate the cost of OXC's of three to five, and six to ten incident fibers.

$$5.35 \times \sum_{j \in S: \mu_j^k=1} F_j + 2, \quad \forall k \in N: 2 < \sum_{j \in S: \mu_j^k=1} F_j \leq 5. \quad (4.21)$$

$$5.85 \times \sum_{j \in S: \mu_j^k=1} F_j + 2, \quad \forall k \in N: 5 < \sum_{j \in S: \mu_j^k=1} F_j \leq 10. \quad (4.22)$$

In cases of more than ten incident fibers, the OXC is duplicated (as much as needed) to handle exceeding fibers. More general equations to compute OXC costs are given by (4.23)–(4.25): equation (4.23) is for cases where there are two exceeding fibers to the overall node, i.e. 12, 22, 32, etc; and equations (4.24) and (4.25) are used in cases of three to five and six to ten exceeding fiber ports, respectively. S , F_j , F_{w_j} and F_{s_j} follow ILP₁₈ symbology; N represents the set of nodes across the network; and $\mu_j^k \in \{0, 1\}$ indicates whether or not node k is an end-node for span j ; ($\mu_j^k = 1$ if it is, and $\mu_j^k = 0$ if it is not).

$$60.5 \times \sum_{j \in S: \mu_j^k=1} (F_j \text{ div } 10) + 11.8, \quad \forall k \in N: \sum_{j \in S: \mu_j^k=1} (F_j \text{ mod } 10) \leq 2. \quad (4.23)$$

$$60.5 \times \sum_{j \in S: \mu_j^k=1} (F_j \text{ div } 10) + 5.35 \times \sum_{j \in S: \mu_j^k=1} (F_j \text{ mod } 10) + 2, \quad (4.24)$$

$$\forall k \in N : 2 < \sum_{j \in S: \mu_j^k=1} (F_j \text{ mod } 10) \leq 5.$$

$$60.5 \times \sum_{j \in S: \mu_j^k=1} (F_j \text{ div } 10) + 5.85 \times \sum_{j \in S: \mu_j^k=1} (F_j \text{ mod } 10) + 2, \quad (4.25)$$

$$\forall k \in N : 5 < \sum_{j \in S: \mu_j^k=1} (F_j \text{ mod } 10) \leq 10.$$

(4.26)

4. Fully Transparent p -Cycle Designs

On the basis of hypothetical whole fiber cross-connect cost reductions in Section 4.2.2, each glass switching device costs 11.8 if there are exactly two (working *plus* protection) fibers incident to the device. Equations (4.27) and (4.28) are used in three to five and six to ten incident fiber cases, as equivalences for (4.21) and (4.22). General forms for equations (4.23)–(4.25) are quite difficult to derive because working and protection fibers do not have the same price. As there is no rule to differentiate between working and protection among exceeding and non-exceeding fibers, two different OXCs (one for working and the other for protection) will be considered in cases where there are more than ten incident fibers.

$$5.35 \times \sum_{j \in S: \mu_j^k=1} (F_{w_j} + 0.1 \times F_{s_j}) + 2, \quad \forall k \in N: 2 < \sum_{j \in S: \mu_j^k=1} (F_{w_j} + F_{s_j}) \leq 5. \quad (4.27)$$

$$5.85 \times \sum_{j \in S: \mu_j^k=1} (F_{w_j} + 0.1 \times F_{s_j}) + 2, \quad \forall k \in N: 5 < \sum_{j \in S: \mu_j^k=1} (F_{w_j} + F_{s_j}) \leq 10. \quad (4.28)$$

Transponder Cards

OXC transponder cards are priced on the basis of three *maximum transmission distances* (MTDs): 750, 1500 and 3000 km. For the current prevalent data rate of 10 Gbit/s, [GLW⁺06, HGMS08] specifically report transponder costs of 1 for MTDs of 750 km, 1.4 for MTDs of 1500 km and 1.9 for MTDs of 3000 km. The same trend will be maintained for transponders needed on the whole fiber cross-connect switch.

Transparent Node Amplifier

One transparent node amplifier is placed at the entrance/exit of every OXC (whole fiber cross-connect device, in case) for each incident bidirectional fiber. The cost incurred is 1.25 a unit; equation (4.29) determines the cost of all transparent amplifiers at a given node k .

$$1.25 \times \sum_{j \in S} F_j \cdot \mu_j^k, \quad \forall k \in N. \quad (4.29)$$

Span Costs: IAs, DCFs, DGEs

In Figure 4.6, the optical signal also goes through an IA, a DCF every 80 km, and a DGE every four IAs. Any IA, DCF or DGE is shared by all optical paths traveling across a specific *bidirectional fiber*. From the CapEx perspective, IA and DCF prices depend on the

4. Fully Transparent p -Cycle Designs

three MTDs: 750 km, 1500 km and 3000 km. An IA costs 3 for MTDs of 750 km, 3.8 for MTDs of 1500 km and 4.7 for MTDs of 3000 km. With respect to the same MTDs, DCF costs are of 0.9, 1 and 1.2. The NOBEL normalized cost for DGE is a constant equal to 3.

Maximum Transmission Distance (MTD)

Table 4.4 summarizes equipment costs in the NOBEL worksheet. Most prices depend on MTDs, meaning that if two transparent lightpaths of 600 and 800 km share a given transponder, the cost of that transponder is defined for an MTD of 1500 km because the next lowest MTD is 750 km which cannot handle the 800 km path. This definition typifies end-to-end paths only; in fully opaque p -cycle designs, the MTD of pure transmission equipment is based on the length of its longest incident span. For hybrid p -cycle configurations, an equipment MTD is based on the longest length of end-to-end working paths and protection path-segments (within available p -cycles) that use the equipment in question. In fully transparent p -cycle designs, switching either at the wavelength or at the fiber level of granularity, the MTD for a transponder is imposed by the longest normal or restored state lightpath using the equipment in question. For IA and DCF that relate to span fibers, two sets of MTDs are considered: one for working fibers and the other for protection fibers. The former is based on the length of the longest working path crossing the span in question while protection MTD depend on the longest restored state path length.

Table 4.4: *Summary of Equipment Pricing Under the NOBEL Cost Model*

MTD	IA	DCF	DGE	Transponder	EXC Port
750 km	$3 \times \lceil \frac{C_j}{80} \rceil$	$0.9 \times \frac{C_j}{80}$	$3.0 \times \lfloor \frac{\lceil \frac{C_j}{80} \rceil}{4} \rfloor$	1	0.28
1500 km	$3.8 \times \lceil \frac{C_j}{80} \rceil$	$1.0 \times \frac{C_j}{80}$	$3.0 \times \lfloor \frac{\lceil \frac{C_j}{80} \rceil}{4} \rfloor$	1.4	0.28
3000 km	$4.7 \times \lceil \frac{C_j}{80} \rceil$	$1.2 \times \frac{C_j}{80}$	$3.0 \times \lfloor \frac{\lceil \frac{C_j}{80} \rceil}{4} \rfloor$	1.9	0.28
6000 km	$5.7 \times \lceil \frac{C_j}{80} \rceil$	$1.5 \times \frac{C_j}{80}$	$3.0 \times \lfloor \frac{\lceil \frac{C_j}{80} \rceil}{4} \rfloor$	2.5	0.28

MTD calculation for hybrid and fully transparent p -cycle designs requires measuring (and limiting) the length of normal state paths, restored state paths and protection path-segments. For test case network instances found too large to respect MTDs of 3000 km, a new MTD value of 6000 km is defined in the last row of Table 4.4, following the trend of

prices from 750 to 1500 km (i.e., 2×750) and from 1500 to 3000 km (i.e., 2×1500); for the MTD of 6000 km, an IA costs 5.7, a DCF costs 1.5 and an OXC costs 2.5.

4.2.4.2 Case Studies and Experimental Results

A series of experiments was aimed at providing real-world CapEx costs for the collection of test case networks. But a direct cost optimization under standardized cost models is a problem of considerable complexity. Instead, the cost evaluation consisted of solving p -cycle ILPs for every configuration type and each network under consideration, and characterizing pre-generated design solutions from the NOBEL cost perspective.

Table 4.5 summarizes experimental results. Test case instances in the first column comprise: **Havana** when considering the original demand pattern with shortest hop-count or distance weighted routing under no-failure conditions, **Cost239**, **Italy**, **Bellcore** and **Euro**. The largest traffic matrix for the **Havana** network is not considered, because in equation (4.16), all units for a given demand relation are required to ride onto the same fibers from end-to-end, where each fiber comprises 40 wavelengths only (i.e. $> d^r$ of 100). The first column of Table 4.5 also indicates the different types of p -cycle configurations. As well, optical paths are often constrained under certain length limits in order to match MTD-related equipment prices. Transparent reach limits of 750, 1500 and 3000 km are imposed in priority; and this is pushed up to 6000 km in cases where optimization results in unfeasibility under primary length limitations.

Spare Capacity Required in the Designs

The second column of Table 4.5 records total spare capacity requirements in prior-generated design solutions. If not indicated, total capacities are expressed in terms of channel-kms for wavelength switched p -cycle solutions vs. fiber-kms for glass switched designs. With **Havana**, spare capacity is expressed in channels and fibers for wavelength and whole fiber switched p -cycles, when considering the original set of demands over shortest hop-count routes. Higher spare capacities are required to achieve designs under smaller optical reach limits. Considering the same reach value, fully transparent wavelength-based protection requires more spare capacity than opaque and hybrids because of optical reach limits imposed to *long* end-to-end fully transparent paths in normal and restored network states. In

4. Fully Transparent p -Cycle Designs

Table 4.5: Sample Results for Overall Real CapEx Costs

Architecture	Min Spare Capacity	Node Cost	Span Cost	Xssion Cost	CapEx Cost	Unused Wavelengths	Used Channels max load span
Havana (hop-count based routing)							
<i>opaque</i>	134 channels	374	225	1035	1634	740 out of 1040	22 for 1 fiber
<i>hybrid</i>	Not feasible for optical reach of 750 km						
- 1500 km	152 channels	374	275	613	1261	722 out of 1040	26 for 1 fiber
- 3000 km	134 channels	374	325	646	1345	740 out of 1040	22 for 1 fiber
<i>transparent</i>	Not feasible for optical reach of 750 km						
- 1500 km	168 channels	686	524	262	1472	1746 out of 2080	32 for 2 fibers
- 3000 km	134 channels	661	545	373	1579	1700 out of 2000	22 for 2 fibers
<i>glass</i>	Not feasible for optical reach of 750 km						
- 1500 km	53 fibers	451	516	269	1236	874 out of 1040	20 for 1 fiber
- 3000 km	45 fibers	457	486	366	1309	874 out of 1040	20 for 1 fiber
Havana (shortest distance weighted routing)							
<i>opaque</i>	20,264	374	225	1224	1823	705 out of 1040	32 for 1 fiber
<i>hybrid</i>	Not feasible for optical reach of 750 km						
- 1500 km	25,513	374	275	663	1312	680 out of 1040	32 for 1 fiber
- 3000 km	20,264	374	310	712	1395	705 out of 1040	32 for 1 fiber
<i>transparent</i>	Not feasible for optical reach of 750 km						
- 1500 km	28,677	686	513	229	1428	1697 out of 2080	33 for 2 fibers
- 3000 km	20,264	647	506	355	1508	1625 out of 1960	32 for 2 fibers
<i>glass</i>	Not feasible for optical reach of 750 and 1500 km						
- 3000 km	6,691 fib-kms	466	460	340	1266	866 out of 1040	16 for 1 fiber
Cost239							
<i>opaque</i>	85,640	342	800	1600	2742	493 out of 960	38 for 1 fiber
<i>hybrid</i>	Not feasible for optical reach of 750 and 1500 km						
- 3000 km	115,290	391	1250	1151	2792	550 out of 1080	52 for 2 fibers
- 6000 km	86,240	485	1703	1117	3305	853 out of 1320	38 for 2 fibers
<i>transparent</i>	Not feasible for optical reach of 750 and 1500 km						
- 3000 km	128,555	631	2113	511	3255	1369 out of 1920	52 for 2 fibers
- 6000 km	86,290	644	2099	674	3417	1293 out of 1760	38 for 2 fibers
<i>glass</i>	Not feasible for optical reach of 750, 1500 and 3000 km						
- 6000 km	18,585 fib-kms	380	1591	657	2628	585 out of 880	37 for 1 fiber
Italy							
<i>opaque</i>	55,654	685	280	5960	6925	523 out of 2320	196 for 5 fibers
<i>hybrid</i>	Not feasible for optical reach of 750 and 1500 km						
- 750 km	90,892	452	369	2974	3795	816 out of 2960	263 for 9 fibers
- 1500 km	55,929	401	472	2860	3732	1417 out of 3200	182 for 6 fibers
- 3000 km	55,654	435	515	2932	3882	1203 out of 3000	196 for 7 fibers
<i>transparent</i>	Not feasible for optical reach of 750 and 1500 km						
- 750 km	120,471	375	527	651	1553	1608 out of 4160	223 for 7 fibers
- 1500 km	58,241	326	501	1026	1853	1756 out of 3560	210 for 9 fibers
- 3000 km	55,654	502	445	1366	2313	1083 out of 2880	196 for 7 fibers
<i>glass</i>	Not feasible for optical reach of 750 km						
- 1500 km	4707 fiber-kms	456	362	1006	1824	501 out of 1440	123 for 4 fibers
- 3000 km	4420 fiber-kms	443	463	1357	2163	501 out of 1440	123 for 4 fibers

4. Fully Transparent p -Cycle Designs

Architecture	Min. Spare Capacity	Node Cost	Span Cost	Xssion Cost	CapEx Cost	Unused Wavelengths	Used Channels max load span
Bellcore							
<i>opaque</i>	14,591	709	160	4500	5369	462 out of 1960	116 for 3 fibers
<i>hybrid</i>							
- 750 km	16,274	524	232	2471	3227	641 out of 2240	164 for 5 fibers
- 1500 km	14,591	457	279	2471	3207	462 out of 1960	116 for 3 fibers
- 3000 km	14,591	457	279	2471	3207	1262 out of 2760	116 for 4 fibers
<i>transparent</i>							
- 750 km	17,784	522	236	843	1600	1245 out of 2880	180 for 5 fibers
- 1500 km	14,591	569	304	1181	2055	1742 out of 3240	116 for 4 fibers
- 3000 km	14,591	564	325	1363	2252	1662 out of 3160	116 for 3 fibers
<i>glass</i>							
- 750 km	1615 fiber-kms	368	225	846	1439	499 out of 1360	100 for 3 fibers
- 1500 km	1270 fiber-kms	395	193	1169	1757	499 out of 1360	100 for 3 fibers
- 3000 km	1259 fiber-kms	399	213	1250	1862	499 out of 1360	100 for 3 fibers
Euro							
<i>opaque</i>	235,206	1416	893	11,815	14,124	835 out of 4680	164 for 5 fibers
<i>hybrid</i>							
- 3000 km	Not feasible for optical reach of 750 and 1500 km						
- 3000 km	260,974	1164	1973	6935	10,072	2670 out of 6760	204 for 7 fibers
<i>transparent</i>							
- 3000 km	Not feasible for optical reach of 750 and 1500 km						
- 3000 km	283,729	1210	1818	4358	7386	1838 out of 6080	190 for 6 fibers
<i>glass</i>							
- 3000 km	Not feasible for optical reach of 750 and 1500 km						
- 3000 km	19,861 fib-kms	740	1790	4404	6934	930 out of 2920	96 for 3 fibers

contrast, other wavelength switched p -cycle configurations force the same distance limits upon end-to-end working paths and *smaller* protection path-segments for hybrids, and on spans for opaque designs. The comparison does not apply to glass switched p -cycles, of which capacity requirements are expressed in terms of fiber-kms (instead of channel-kms).

CapEx Cost Perspectives

In columns 3 to 6 of Table 4.5, CapEx costs are broken into four data sets: node, span, transmission and overall total costs. Node costs encompass the costs of OXCs (or whole fiber cross-connects, in case) with their fiber ports and amplifiers. Span costs include IA, DCF and DGE elements that are allocated per bidirectional fiber span. Transmission costs encompass all path termination and per-signal switching costs, i.e. EXC ports and transponder cards.

In opaque networks, node costs used to be low because of the flexibility of opacity that implies less fiber requirements and thus less OXC fiber ports, in comparison with other

4. Fully Transparent p -Cycle Designs

configuration types. Spans are cheaper as the number of IAs, DCFs and DGEs involved in the equipment pricing exercise decreases with the smaller number of fibers required in the design. Furthermore, IAs and DCFs are applied the lowest possible costs because MTDs in opaque architectures, where wavelength conversion assumed at all nodes are imposed by longer spans. But transmission costs are much higher with opaque p -cycles than with any other configuration type because of EXC ports and transponder cards required for electrical switching operations. Due to those higher transmission costs, opaque p -cycle designs appear more expensive than others, despite node and span cost-effectiveness.

Although not as expensive as opaque p -cycle configurations, hybrid and transparent wavelength switched p -cycle designs are still a bit expensive. Node costs are cheaper with hybrids due to a lower number of fiber requirements in the design. But span costs are higher than in the opaque case as they are subject to higher MTDs, calculated on the basis of longer end-to-end working paths and protection path-segments. More harmful, transmission costs significantly rise up (once again) because electrical switching at p -cycle entry points increases the number EXC ports and OXC transponders. In contrast, fully transparent wavelength switched p -cycles involve lower transmission costs, due to the optical switching (i.e., one transponder) at the sole ingress and egress nodes for add/drop operations. But this is subject to some node and span cost penalties because more fibers (one set for working and another for protection purposes) are used in the design. Also, MTDs are much higher as they are calculated on the basis of longer (end-to-end) normal and restored network state path lengths.

Which of the hybrid or fully transparent wavelength switched p -cycle configuration is better from CapEx cost perspectives? The answer depends on whether or not the network under consideration is lightly or highly loaded. For instances with lower traffic volumes, such as **Havana** and **Cost239**, protection switching is seldom required in the hybrid type of configuration; resulting EXC port and transponder card costs cannot balance with node fiber port, IA, DCF and DGE equipment prices in corresponding fully transparent p -cycle designs. In contrast, highly loaded networks such as **Italy**, **Bellcore** and **Euro** involve much lower costs in fully transparent p -cycle contexts. The penalty imposed to transmission costs by hybrid intermediate protection switching is much higher than that of fiber duplications to node and span costs required by full transparency.

Whole fiber switched p -cycles stand as the most promising configuration from a CapEx perspective. The fiber-level protection paradigm greatly diminishes the number of fibers in the design. This results in lower node costs, somewhere in between that of opaque/hybrid and fully transparent wavelength switched p -cycle designs, as the set of protection fibers involve as well an additional 90% saving over working fiber port costs. Span and transmission costs are quite similar to that of fully transparent p -cycle designs because of higher MTDs, based on longer end-to-end normal and restored state path lengths. More generally, whole fiber switched p -cycles appear to be of great benefits to overall CapEx costs for a wide range of networks and traffic payloads. In practice, glass switched p -cycles are *applicable to metro-core networks*, which are basically cost sensitive and highly loaded.

Fiber Utilization Ratios

The seventh and eighth columns in Table 4.5 pertains to fiber utilization levels. They specifically report the ratios of unused to total wavelengths and the numbers of fibers and wavelengths used on the maximally loaded span. In contrast to wavelength switched p -cycles, results for whole fiber switched p -cycles only address *working fiber* loads (as protection fiber utilization levels follow automatically). Maximally loaded spans show medium to very high fiber utilization ratios: 27 to 95% in the *Havana* and *Cost239* lightly loaded networks and 58 to 98% in *Italy*, *Bellcore* and *Euro* test case instances. In contrast, highly loaded networks show overall small levels of fiber utilization: 51 to 85% of unused wavelengths for lightly network instances and 18 to 54%.

But fiber utilization ratios used to be lower in wavelength and glass switched fully transparent designs, where working and protection wavelengths travel onto different sets of fibers. In the specific case of whole fiber switched p -cycle designs, low fiber utilization ratios can be of great interest for dynamic traffic considerations, through the concept of protected working capacity envelope (PWCE). With as much as 32 to 84% of unused wavelengths available within working fibers, there is a considerably high probability to build working lightpaths within unused wavelength channels for new demand relations. And if so, newcomers are automatically protected as is the philosophy of fiber-level protection.

Optical Reach Concerns

The effect of distance limits on overall CapEx costs is skipped for opaque p -cycle designs because electrical switching is performed at all nodes and the resulting MTDs depend on span lengths only. Most of the time, hybrid p -cycle designs are achievable for all of the three basic MTDs due to regeneration applied at p -cycle access points. This greatly reduces span costs so that in lightly loaded networks, such as *Havana* and *Cost239*, total costs are better than those incurred by corresponding fully transparent wavelength designs, and very close to the total cost of whole fiber switched p -cycles. In fact, fully transparent p -cycle design types are achievable for higher MTDs, of which calculation takes into account the length of surviving portions of failed working paths *plus* protection segments. Typically, the conclusions of the CapEx study are valid for the same specific optical reach limit, for all p -cycle configurations of a given network instance. When considering two different optical reach limits, the smaller of them provides (in most cases) minimum CapEx costs regardless of the configuration type under consideration.

4.3 Closing Discussion

It is desirable to have design control over normal working and restored state path lengths in a transparent optical network. An obvious approach with p -cycles is to limit the maximum allowable circumference of candidate cycles considered in the network design. But this is somewhat inefficient and does not directly control the end-to-end length of paths in a restored state; it only controls the maximum length of protection path-segments that might be substituted into a working path on failure. Another basic strategy considered in this chapter consists of systematically matching shorter working paths with longer protection path-segments through p -cycles, and vice versa, with direct consideration of the end-to-end length of paths in the restored network state. This complementary matching notion was studied through design models minimizing spare capacity while intelligently associating longer working paths with shorter protection path-segments and vice versa. The basic p -cycle ILP was adapted in one case to minimize the average restored state path lengths, in another to achieve the least possible longest path length, and finally to constrain all restored path lengths under a fixed limit. Each variation was subject to a requirement

4. Fully Transparent p -Cycle Designs

of using only the theoretically minimal spare capacity or, through bi-criteria methods, a minimal amount of additional spare capacity for the corresponding objective on path lengths. Taken overall, this study provides the means to design an entire transparent survivable island that respects the transparent reach limits of a given ULH technology.

From another perspective, the cost and complexity of wavelength assignment and conversion and wavelength-selective switching are always primary considerations in the design of survivable optical networks. And yet, while nodes and single DWDM channels may fail, a pre-dominant source of unavailability is physical damage to optical cables. If it is ultimately glass that fails, what if just the glass is directly replaced? More specifically, what if p -cycles were used to rapidly, simply and efficiently provide for the direct replacement of failed fiber sections with whole replacement fibers? As long as the loss budgets are adequate, entire DWDM wavebands could be restored with no switching or manipulation of individual lightpaths, so that the DWDM layer would never know the break happened. Environments where fiber switching devices are low cost, and ducts are full of dark fibers provide a very low CapEx cost alternative to protect an entire DWDM transport layer (or working capacity envelope) against the single largest cause of outage.

Despite many benefits, p -cycle protection at the glass fiber level might show several disadvantages. The principle only acts in the event of span failures in the optical transmission layer; a priori, neither OXC nodes nor upper-layer routers (e.g. IP/MPLS routers) can be protected against failures through this principle. On the other hand, the failure of one wavelength into a given fiber requires the switching of all traffic carried on any wavelength within the fiber in question. One may also experience some limitations on glasses in the design because the number of wavelengths per fiber maybe not enough to provide the capability. But all of those disadvantages of glass switched p -cycles are only a priori thoughts, which require further research on the topic. Hopefully, this contribution increases interest on the overall question.

Chapter 5

A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

The present chapter is adapted and extended from [GO08, OG08a, OBG09, OG10b, OG10a], which deal with large scale p -cycle network design problems. The aim of the contribution is a novel combination of genetic algorithms (GA) and integer linear programming (ILP), referred to as GA-ILP, which seems to have many features to recommend it for any large scale p -cycle network design problem involving the preselection of a relatively few candidate cycles, from a practically infinite set of all possible cycle structures.

Section 5.1 discusses large scale issues in the planning of p -cycle networks, and acknowledges prior work that may lead to practical methods for overcoming those issues. Section 5.2 introduces the GA-ILP and evaluates its reference performance vis-à-vis other methods on smaller test case networks. In further considerations, Section 5.3 addresses a 200-node challenge case instance of the conventional p -cycle design problem; and Section 5.4 shows how the GA-ILP framework serves for the advanced p -cycle design of medium size networks.

5.1 The Large Scale p -Cycle Design Problem and Solutions

A typical p -cycle network design solution comprises the most efficient combination of cyclical structures selected from the space of all candidate cycles. Basically, this suggests an explicit enumeration of all possible cycle structures available within the network graph to be designed, followed by the completion of an ILP optimization model defining the p -cycle problem in question and using the prior-enumerated cycle structures as the candidate set. Cycle pre-enumeration and ILP optimization steps complete quite quickly and easily for

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

a wide range of networks and various p -cycle design problems. For example, the DFS all-cycle finder [Dou01] followed by ILP₁ for conventional p -cycle minimum spare capacity design model reach full optimality in less than to a few running seconds when applied to small and medium size network instances such as Havana, Cost239, Italy, Bellcore and Euro test cases, used for experiments in previous chapters.

But any network optimization problem becomes challenging or even impossible to solve on a very large scale because it involves a huge number of variables and constraints, which possibly exceeds memory limits for the experimental environment or the maximum number of variables and constraints supported by the ILP solver in use. In the specific case of p -cycles, the number of variables and constraints grows exponentially with the number of candidate structures available within the network under consideration. And the rate at which the number of candidate cycles rises is $O(2^{|S|-|N|})$, where $|S|$ and $|N|$ represent span and node numbers respectively. Consequently, ILP₁ has essentially never been attempted on a 200-node network even though the sheer scale of the original p -cycle minimum spare capacity design problem is intrinsically not as complex as optimal network planning models for most other protection architectures (especially path-oriented schemes). In the same manner, p -cycle design models integrating such advanced concerns as controlled optical path length in the restored network state, path-protecting p -cycles or maximum node failure protection were found to be unsolvable in Chapters 3 and 4, even for the collection of medium size test case networks. Subsequently, one may distinguish between three classes of p -cycle design problems based on:

- i. whether the space of all distinct simple cycles is entirely enumerable and solvable,
- ii. fully enumerable but impractical to import into the ILP solver,
- iii. or not even enumerable.

Solving problem instances of the 2nd and 3rd classes requires some advances in the art of p -cycle design.

5.1.1 Preselection Methods

An obvious way to tackle the large scale p -cycle network design problem is to preselect a *sample* of the true space of cyclical structures and consider this specimen alone as the

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

candidate set in the design. In [GD02], the preselection criteria is referred to as cycle *a priori efficiency* and set equal to the total amount of working capacity the cycle structure in question has a potential to protect, divided by the total amount of spare capacity required to build the structure itself. Equation (5.1) gives a mathematical definition for cycle p a priori efficiency (within ILP₁ symbology).

$$AE(p) = \frac{\sum_{i \in S} x_i^p}{\sum_{j \in S: x_j^p = 1} C_j}. \quad (5.1)$$

With (5.1), candidates showing more straddling spans relative to their size (or capacity requirement) have higher potential efficiency as p -cycles. But in a typical network, working channel counts differ on each span; so a p -cycle design solution generally involves structures of different circumference-sizes, not only cycles of high purely topological efficiency. Accordingly, equation (5.2) states that the *actual efficiency* of a p -cycle depends not just on the number of on-cycle and straddling spans but also on working capacity channels to be protected on spans [DHGY03].

$$E_w(p) = \frac{\sum_{i \in S} w_i \cdot x_i^p}{\sum_{j \in S: x_j^p = 1} C_j}. \quad (5.2)$$

5.1.1.1 The CIDA Preselection Heuristic

A well-known preselection heuristic is the *capacitated iterative design algorithm (CIDA)* in [DHGY03], which calculates the actual efficiency of each candidate cycle using equation (5.2) and selects one channel-copy of the candidate with the highest efficiency as the one to be placed in the network. Working capacity values are then updated across the network, by subtracting one working capacity channel from each on-cycle span and two working channel units from each straddling span, in regard to the p -cycle just placed. And CIDA iteratively recalculates $E_w(p)$ for each candidate, places one copy of the cycle structure showing the best efficiency for the design in progress, and updates working capacity values until no more traffic remains unprotected on any span across the network.

Checking on huge numbers of candidate cycles at each iteration is exhausting; especially if only a few dozen are typically involved in the final solution. So CIDA does not run on the entire space of candidates but uses the straddling link algorithm (SLA) in [ZY02] with

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

some enhancements aimed at improving the quality of preselected candidates. With the SLA principle, any cycle finding algorithm such as DFS is called to generate one *primary cycle* for each span across the network; the primary cycle for a given span refers to the shortest cycle straddled by the span in question. As this policy may not cover degree-2 nodes, a *test-and-repair* procedure is recommended by the SLA completion to ensure 100% span coverage, i.e. full span failure restorability.

Unlike pure SLA, CIDA lightly enlarges the enumerated set of primary cycles using either *expand* or *grow* operations in [DHGY03]. The *expand* algorithm considers each primary cycle at once and transforms as many on-cycle spans as possible in straddlers by searching, for each given on-cycle span, a distinct route connecting its end-nodes while being node-disjointed from the cycle under transformation. If such a route is found, the span is removed from the cycle under transformation and replaced by the route just found. And all cycles or only the last (if wanted) formed from any given primary cycle, as progressing with the *expand*, are retained as candidates to the design. The *grow* algorithm is only different in that cycle expansion procedure is reinitialized for the newly obtained cycle, every time a route is found and added to the current cycle.

5.1.1.2 Benefits and Narrowness of CIDA and Other Preselection Heuristics

As CIDA, most preselection heuristics are of greater conceptual simplicity; they do not go through exhaustive all-cycle finding processes; and they greatly reduce the space of candidate structures eligible in the design, giving rise to p -cycle network solutions on very large scales, in a few seconds. But preselection methods still require cycle enumeration, at least up to a circumference-size limit. Network solutions are only *suboptimal* because preselection criteria do not often reflect the exact goals of the p -cycle problem under consideration. Instead, they choose cycles on the basis of their individual merit, rather than looking for combinations of p -cycles working *well together*. In advanced p -cycle problems of more complexity than the conventional p -cycle minimum spare capacity network design problem, preselection methods require subtlety to identify properties for *good cycles*. This need for effective preselection criteria is justified by SLA itself, as it was initially developed on the basis of a priori efficiency in equation (5.1) but has since been proven completely inefficient from the spare capacity requirement perspective [ZY02].

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

TRLabs-pCycle-SCP-CIDA.exe proprietary software in [DG03] will be used to provide CIDA comparative results for first and second problem class instances. At the 200-node sheer scale, this software was unable to provide a CIDA solution after fifteen running days.

5.1.2 Unifying Enumeration-free ILPs within the Transportation-Flow Problem Structure

In contrast to preselection methods, other approaches to the large scale p -cycle design problem directly integrate cycle forming constraints into ILP themselves. The key idea is to *impose on each node across the network to have either 2 or 0 incident spans on the cycle structure in question*. So any on-cycle node for a specific structure will have exactly two incident spans on the structure whereas off-cycle nodes will have zero. This cycle forming strategy belongs to the *flow conservation* principle for *transportation type problems*, and unifies enumeration-free ILP and column generation (CG) approaches for p -cycle design.

5.1.2.1 [Sch04] p -Cycle Enumeration-free ILP

Enumeration-free ILPs follow a line of work by [Sch04], which first generates an index set for a given maximum number of distinct cyclical structures allowed in the design solution (i.e. $1, 2, 3, \dots, |P|$). One single simple cycle is built per index number by requiring flow conservation at every on-cycle node. To avoid multiple cycle structures for the same index number, a master node is assigned to each index and thus to the corresponding cycle; and a path is determined from the master node to every other node across the network, by requiring (once again) flow conservation but at intermediate nodes along the path. If the target node is also part of the cyclical structure, there will be a full on-cycle route to connect it to the master node. As well, [Sch04] avoids non-linear formulations and defines span protecting relationships as going with cycle generation.

The numbers of variables and constraints in [Sch04] model is much lower than in the two-step ILP approach, depending on how many distinct cycle structures are allowed in the solution. Nevertheless, [Sch04] faced computational time issues and getting solutions was only possible through a four-step heuristic. But even with those arrangements, experimental run times of about 4 hours were reported for network cases of 5 to 25-nodes,

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

considering a maximum of $|P| = 5$ cycle structures allowed in final solutions which is even lower than the average dozen typically reported for p -cycle design solutions.

5.1.2.2 Enumeration-free ILP Design Models in [WYH07]

[WYH07] states that very long running times in [Sch04] are due to the sets of constraints assigning one single cycle per index number and a master node to each cycle, whereas any node may take the role. Subsequently, [WYH07] no longer associates each index number with a single cycle alone but let the ILP free to generate multiple cycle structures instead. He defines each on-cycle (straddling) span for any cycle structure in the set as being an on-cycle (straddling) span for the corresponding *index domain*. So if the set of cycles for a given index number is involved in the solution, all on-cycle and straddling spans (relative to any cycle structure in the index domain) are covered.

To check whether or not a given span can be protected by a specific cycle set, [WYH07] proposes three enumeration-free ILP variants. One of them is referred to as *recursion-based ILP* because it checks the connectivity of two nodes using a recursive process. Another *flow conservation based ILP* fully reproduces the transportation-flow problem structure, under the constraint of no more than one unit flow riding on each span. The third and most efficient ILP in the series, referred to as *cycle exclusion based ILP*, recognizes that considering a unique cycle per index number really simplifies the routing problem; it thus maintains the simpler multi-cycle enumeration per index number but also applies a cycle-exclusion based principle to keep only one single per index domain in the design.

The series of enumeration-free ILPs in [WYH07] significantly reduces the complexity of these kinds of problems in terms of variables and constraints. They do not require extra heuristics for solving and they run much faster than the p -cycle design model in [Sch04]. They have been successfully tested on a very large network instance of 30 nodes and 62 spans, for 13,343,782 possible cycle structures. For a requirement of $|P| = 7$ distinct structures in the solution, the cycle exclusion ILP in [WYH07] generated only 115,320 candidate cycles, meaning less than 1% of overall space possibilities; and the network solution was obtained in 8.76 hours with a gap of optimality of 4.83%. However, it was

demonstrated¹ that both the run times and the gap of optimality were much greater than those experienced with CG.

5.1.3 The Column Generation Approach

[ST04, ST05] introduced a CG algorithm for solving the conventional p -cycle design problem, but for joint optimization of working and protection capacity (JCP). In the context of spare capacity placement alone (SCP), CG requires an initial candidate set of one *dummy* cycle structure p for each span i across the network, with an ability to handle the span in question but so expensive that it will never show up in the final solution. The next step consists of solving the *primal* or *master* problem, which is a linear programming (LP) relaxation of the model under consideration—here in, ILP₁ for conventional minimum spare capacity design but with variables s_j and η^p assumed continuous (instead of integer). Every span within the network is assigned a *dual variable* π_j measuring the difference between the span’s actual needs, i.e. the number of working channels on that span, and the provision of protection channels offered to the span in question by p -cycles involved in LP₁ design solution. The measure of unused spare capacity π_j defines the *reduced cost* of potential candidate cycles for another ILP model, referred to as the *dual* or *pricing* problem and aimed at building a new cycle structure that shows the minimum possible reduced cost for use in the primal. If running the dual problem results in a negative minimum reduced cost, the cycle found by solving the dual problem is added to the set of candidates. The primal problem is solved again with the updated candidate set; unused spare capacities on edges are recomputed; the dual problem is solved again; and the three steps are repeated iteratively until the objective function of the dual problem returns a positive reduced cost.

At this point, CG has converged; the final solution is the most recent design obtained by solving the primal problem; and the gap of optimality is given by the difference between objective function values of primal and dual problems at the convergence. CG reports typically claim 1% gaps of optimality and fast run times, which may be promising for the large scale p -cycle design problem. But in the literature, CG seems so far best suited to find high quality solutions on such inherently difficult problems as failure independent

¹In collaborative work between the authors of [WYH07] and the group of Professor Brigitte Jaumard from Concordia University in Montréal, a leader in CG research for optical networks in Canada.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

path protecting (FIPP) p -cycles and p -cycle based protected working capacity envelope (PWCE), rather than fast solutions for very large problem instances [JRBG07, JSR09]. Another practical observation about CG is that to obtain the capability, the amount of development and coding time to achieve an effectively working heuristic for a given problem is a significant initial investment cost. This is because the dual problem requires a subtle definition of the reduced cost; and it comprises as well a set of quadratic constraints as in the *quadratic selective travelling salesman problem*, which eliminate sub-tours while forming a cyclical structure [TS03].

In this thesis, the CG method is considered for comparison purposes; related solutions were kindly provided by Professor Brigitte Jaumard from Concordia University in Montréal and her former PhD student, Dr. Caroline Rocha.

5.2 Towards an Exclusive Combination of GA-Methods and ILP to Solve the Large Scale p -Cycle Design Problem

In reviewing past approaches and other work that may lead to practical methods for solving the large scale p -cycle design problem, the great simplicity and fast run times of the preselection leaning were really tempting. But preselection techniques lack optimality because they choose candidate cycles on the basis of individual merit. This research addresses whether it is possible to bring the efficiency of more complex enumeration-free ILPs and CG heuristics into preselection methods while still retaining the simplicity of those preselection techniques, making them attractive and accessible to the average user.

5.2.1 General Understanding of the GA-ILP Philosophy

Evolutionary meta-heuristics may help to overcome the narrowness of prior preselection methods, as they comprise particular attributes to *recognize combinations or collections of elements working well together*. More specifically, genetic algorithms (GA) must be effective at reducing a vast space of cyclical structures to a much smaller but highly relevant subset of candidate cycles; because GA's nature is to sample problem spaces widely while accumulating and promoting good features, good design ideas and promising options within a genome. But the efficiency of a GA-based preselection is only guaranteed by an effective

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

fitness function, capable of reflecting as much as possible the goals of the original problem statement, not only such preselection criteria as a priori or actual efficiencies.

Knowing that integer linear programming (ILP) is truly good at selecting and combining design elements and at reflecting many details for a given problem, what if a GA evolves a preselected population for a subsequent ILP solution? Furthermore, what if the fitness function of the GA is the objective function of the ILP itself!? These two fundamental questions gave rise to a novel combination of GA-methods and ILP, now referred to as GA-ILP. The GA-ILP philosophy is to sample the wide space of all simple distinct cycles available within a network, by letting GA accumulate high-merit options and solution elements from that space and allowing a final fully detailed ILP assemble the best combination of elements from the GA-preselection and for the given objectives, using the original p -cycle model itself or a similar ILP problem as fitness function for the GA.

The GA-ILP philosophy is very different than prior unsuccessful attempts to use meta-heuristics for solving p -cycle design problems, in the sense that forming cyclical structures is no longer the goal pursued. Rather, the GA-ILP conceptually relates to preselection techniques that work on an already existing cycle space. The difference is that combinatoric synergies are not missed as when cycle structures were individually assessed albeit it with very fast and simple filtering criteria. Thus, the GA-ILP still requires an all-cycle finding procedure. But nevertheless, the DFS all-cycle finding algorithm can enumerate hundreds of thousands of cycle structures in less than to few seconds. Considering them as a candidate set, ILP₁ optimization completes in less than a second to a few minutes; so it is not unreasonable to consider a p -cycle ILP as the fitness function for a GA.

If the GA-ILP works well as will become apparent we hope, it will have the attraction of greater conceptual simplicity, common to most preselection techniques within the literature. The GA-ILP will also represent an entirely repeatable meta-heuristic that simply re-uses existing p -cycle ILP models. This is a significant advantage over the problem-specificity of preselection criteria and CG reduced costs, and over the complexity apparent in enumeration-free ILPs that will certainly increase with more *detailed* p -cycle problems. Almost any network size will be possible within the GA-ILP framework but the ILP solver will never face a horrendously large and impractical sub-problem instance because (individuals are) of controllable size through GA-aspects.

5.2.2 GA-ILP Evolutionary Steps and Programming Aspects

Figure 5.1 pictures the GA-ILP in practice. To state it in a summary form, a p -cycle ILP design problem is to be solved but is too large to run with all possible cyclical structures considered as the candidate set P . However, the same ILP or potentially a surrogate “cousin” ILP can be used as fitness function for a smaller subset of candidate cycles that comprise an individual. The hypothesis is that the normal steps of a GA-like iterative process will evolve a still suitably small overall population. The union of individuals from this population embodies a preselected subset of candidate structures on which a final ILP, only somewhat smaller instance of the original problem, can very likely be solved to the same solution quality that could be achieved had the problem been solvable in the first place with the entire set of candidate structures. With this conceptual overview in mind, the following provides details of the novel GA-ILP.

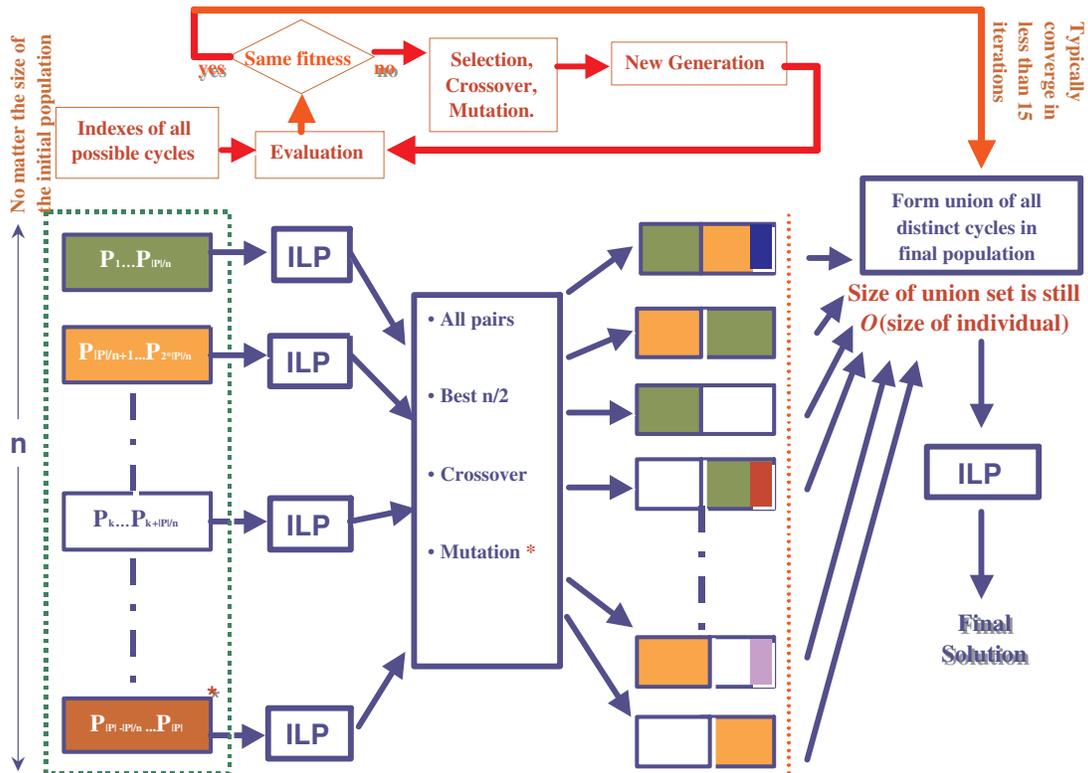


Figure 5.1: The GA-ILP Preselection Heuristic

5.2.2.1 Encoding

The set P of all possible candidate cycles available within the graph is first partitioned into n subsets of $|P|/n$ nominally equal number of cycle structures. Each subset comprises an *individual* of which the genome is the index numbers of cycle structures constituting the subset in question. And the union of all individuals embodies an *initial population*, which is to be improved through GA-like iterations for a specific p -cycle network design problem. Neither the initial population size nor the primary allocation of cycles per individual will affect the quality of the GA-ILP preselection or the final network solution. The idea is just to balance n so that there are the most individuals in the initial population, each easily *evaluated* by the GA constituent ILP without resulting in infeasibility. So if there was no solution for one or more of the above individuals, the ILP solver will quickly report it and the partition of P will be reinitialized with fewer slightly larger candidate subsets.

Considering an instance of ILP₁ for conventional p -cycle minimum spare capacity design, the space P of all distinct simple cycles can be partitioned in practice as follows. On-cycle and straddling protection relationships are first identified between each possible cyclical structure and every span across the network. And P is partitioned so that each subset of candidate structures, comprising an individual of the initial population, is capable of handling all spans within the network; this is in order to guarantee full restorability in the event of any single span failure scenario. So setting n less than or equal to the number of cycle structures handling the span covered by the least number of candidates is well suited to guarantee the feasibility of ILP₁ for any individual of a subsequent initial population. Experimental results further address questions of population sizes.

5.2.2.2 Evaluation

The objective function value of the p -cycle ILP to be solved represents the *fitness function* for the GA-like evolutionary process. This means to evaluate a given population, sub-instances of the p -cycle ILP in question are solved, using in turn individuals of the current population as the candidate set. Every individual is assigned a weight equal to the objective function value of its optimal solution. Considering for example the conventional p -cycle minimum spare capacity design model, the weight of a given individual will be the least spare capacity required to achieve a 100% span restorable design, obtained by solving ILP₁

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

with cycles forming the individual in question considered as the candidate set. In case of infeasibility, the weight of the individual just solved will be set equal to a very high or very small number, depending on whether or not the ILP under consideration is a minimization or a maximization problem. This measure is taken because even though the existence of p -cycle design solutions is guaranteed for each subset of the initial partition of P , evaluating individuals of subsequent populations may *unexpectedly* result in infeasibility because of such reasons as crossover or a deleterious mutation policy (to follow).

Relatively, the ILP fitness function is not complex to evaluate. Whereas the space of all cycles is too large to be imported to the ILP solver, individuals comprising smaller subsets of candidates give rise to rather fast and easy evaluations because of controllable size.

5.2.2.3 Selection and Crossover

Thus, individuals for the GA are candidate cycle subsets and the GA fitness function is the objective function value of the p -cycle ILP under consideration. The $n/2$ best pairs of individuals are retained for *crossover* following evaluation. If the ILP in question is a minimization (maximization) problem, the smaller (higher) the sum of the weights of two parents is, the better that parent pair is for reproduction. Every parent pair selected for breeding produces two children by crossing the first half of one parent's genome (i.e. cycle index numbers) with the second half of the genome of the second parent, and vice versa.

A specific aspect of the GA-ILP developed here is that following crossover, all actual *solution* p -cycles of the individuals not selected for breeding are still recorded to be used as raw material for mutation. The reason is that those cycles may embody some attributes of merit, given that they were selected by the solver as part of the optimal solution for the ILP sub-problem represented by the individual. So an individual may consist of 118 candidate cycles and its ILP₁ solution may be based on a specific dozen p -cycles (typically). If the spare capacity value of the individual is not low enough to rank it for reproduction, it will not continue into the next generation as a parent, but its specific solution cycles will be kept and used to strengthen and propagate genetic diversity in the offspring.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

5.2.2.4 Mutation

In the specific mutation policy here, unselected individuals of the previous generation ($t-1$) with solutions of (say) very high spare capacity requirements are considered in turn. Cycle structures that made up their solutions are inserted into a randomly chosen *child* of the current iteration t . And for every candidate structure inserted in this way, another cycle index is randomly removed from the child such that its total number of cycles remains the same. This may of course be deleterious to the individual but the philosophy is that these cycles may embody some meritorious design elements given that they arose as solution p -cycles, albeit from individuals that did not go on to reproduce.

Note that the mutation policy affects at most half of the offspring because at least one half of the n individuals are required to find $n/2$ parent-pairs, for breeding, leaving no more than half of the individuals of the population for mutation. As well, because of the *random* substitution of a few cycles in several individuals as part of the mutation policy, two experiments running under the same conditions may exhibit slightly different objective function values, candidate preselection sets and final solutions.

5.2.2.5 Terminating Conditions

The generational process (i.e. evaluation, selection, crossover and mutation) is repeated until all the individuals of a given generation have nearly the same fitness. In other words, GA iterations are stopped when every individual in the population has the same ILP objective function value, within the MIPGAP being employed.

The union of all unique cycles of individuals comprising the most recent population represents the GA-ILP preselected set of candidates, to be presented to the final fully detailed ILP, the termination of which yields the final p -cycle network design solution. Note that the union may lead to a small upward creep in the size of the set of candidate cycles, in comparison with the size of the individuals, given that individuals of the last generation are not necessarily identical although having the same fitness. In discussing experimental results below, details of this practical concern will be further addressed.

5.2.2.6 Implementation Aspects

The GA-ILP is a meta-procedure that requires no custom programming, but can just use any ILP solver instead. In experiments to follow, it was entirely implemented in AMPL 10.1 and solved using CPLEX 10.1.0 with a MIPGAP of 10^{-3} , on a four-processor Sun UltraSparc III running at 3GHz with 16GB of RAM. TR Labs preparatory software for initial routing and candidate pre-enumeration was run on a 2.8GHz Intel Core 2 Duo processor with 4GB 1067MHz DDR3 running Mac OS X version 10.5.8.

5.2.3 Effectiveness and Performance of the GA-ILP

As an initial exploration of the GA-ILP capabilities, a series of experiments was conducted with the purposes of studying the convergence over iterations, analyzing the influence of population sizes n , and checking on the growth of final preselection sets. With the conventional p -cycle minimum spare capacity design model defined as GA constituent ILP, experiments were conducted on two network instances from problem classes 1 and 2, respectively. **Euronet** in Figure 5.2(a), from the European community project model, consists of 19 nodes and 40 spans and characterizes the primary class of problems in the sense that the full space of 84,963 candidate cycles is enumerable and an optimal network solution for ILP_1 is reachable. Thus, the GA-ILP equivalence of ILP_1 is not strictly required to obtain a solution; but nonetheless, it will still be applied and compared to the reference solution in order to assess GA-ILP capabilities. The other test case network is **Cse1t** in Figure 5.2(b), an instance of the second problem class which consists of 30 nodes, 56 spans and up to 387,740 distinct simple cycles. This wide space of candidate cycles is also fully enumerable in practice but once imported into the solver, it exceeds the 16 GB memory limit of the experimental environment. Because the ILP solver is unable to support ILP_1 instance for **Cse1t**, providing a p -cycle design solution for this network will be a straightforward application of the GA-ILP, the kind of problems for which the overall GA-ILP approach is intended. Traffic matrices were generated for **Euronet** and **Cse1t** on a basis of node-pairs, following a uniform distribution on the interval [0..10]. Under normal network states, a single least hop working path routing was applied, resulting in 1968 and 4159 working channels to be protected in **Euronet** and **Cse1t**, respectively. The ILP_1 reference solution

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

for **Euronet** requires 834 spare capacity channels, for a redundancy of 42%; whereas (for now) no benchmark of comparison exists for **Cselt**.

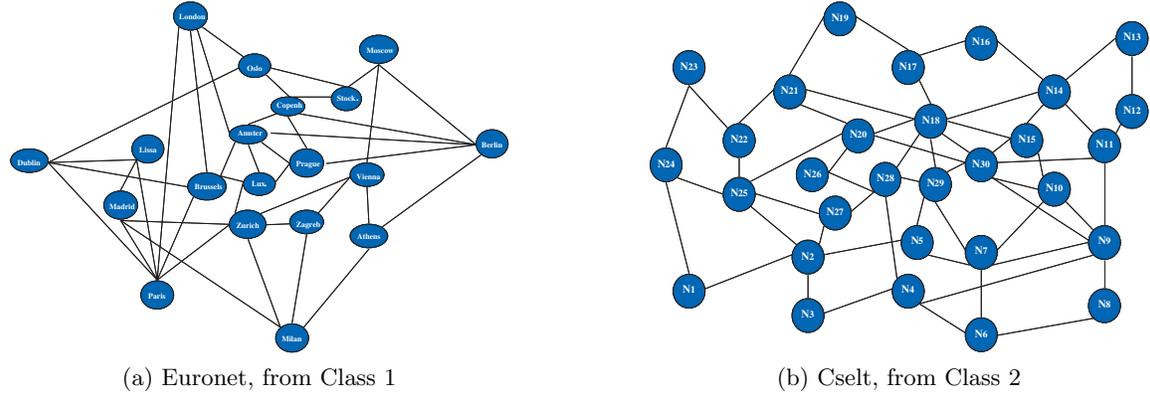


Figure 5.2: *Large Scale Network Instances*

5.2.3.1 Detailed Analysis of the GA-ILP Convergence

Figures 5.3 show the GA-ILP convergence over iterations, in planning both **Euronet** and **Cselt** networks, considering populations of $n = 250$ individuals. The x-axis numbers generations while the y-axis gives, for each of those iterations, three spare capacity measures—i.e. minimum, maximum and average spare capacity requirements—comparing optimal solutions obtained with individuals of current generation considered as the candidate set to the design. As an optimal solution was obtained for the **Euronet** instance of ILP₁, the curves in Figure 5.3(a) give the percentile of extra spare capacities over min-cost requirements. A semi-logarithmic scale is used for convenience, with the x-axis crossing the y-axis at 0.01 (as there is no zero log value); so any percentage of extra spare capacity of 0.01 on the plots is in fact equal to 0. In contrast to **Euronet**, the **Cselt** network has no benchmark for comparison *for now*; the curves in Figure 5.3(b) give the real number of spare capacity channels required by optimal solutions of the individuals in question.

The GA-ILP converges when plots for minimum, maximum and average spare capacity requirements intersect; as this occurs at the iteration for which all individual optimal solutions have the same objective function value within the MIPGAP being employed. Accordingly, the GA-ILP₁ reached the point of convergence in the fifteenth generation

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

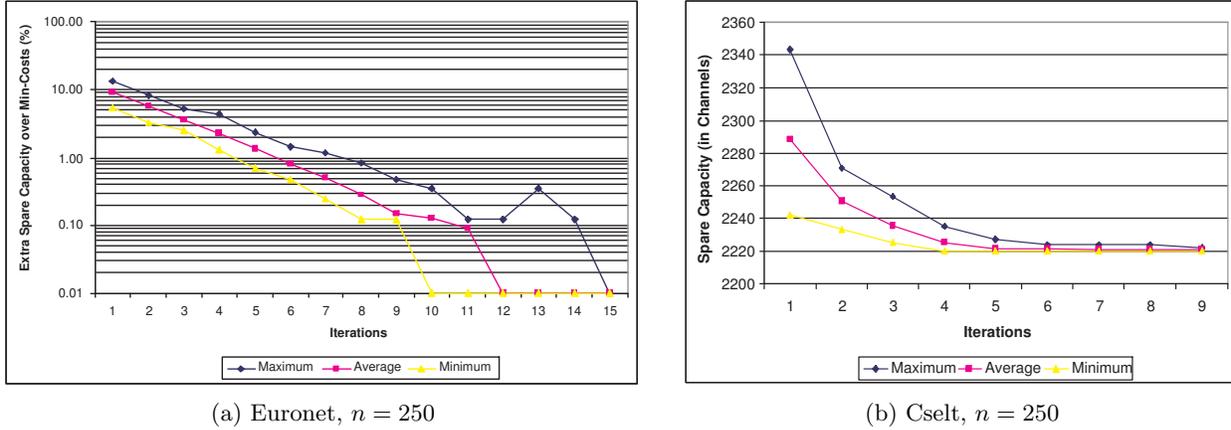


Figure 5.3: Convergence of the GA-ILP for a Given Population Size.

for **Euronet** and the ninth for **CselT**. The **Euronet** case for which an exact solution is known converged to optimality, i.e. with no additional spare capacity relative to ILP_1 reference solution. Significantly, individuals themselves arrived (near-)optimality, meaning that rather unexpectedly but understandably in hindsight, the common objective function value at the convergence is also the objective function value of the final solution obtained by solving ILP_1 with the final merged population. This suggests that any of the individuals of the last generation may constitute the GA-ILP preselection set. But as will be seen for multi-criteria optimization, there might be other good reasons to consider the union of all unique cycles comprising individuals from the last iteration.

5.2.3.2 Influence of Population Sizes on the Convergence and Solution Quality

In addition to $n = 250$, the series of experiments was conducted on the basis of many different values for population size (i.e. $n = 10, 100, 250, 500, 750, 1000, 3000$). Figure 5.4 reports maximum spare capacity requirements for individuals of each given generation, not the minimum or the average relative to previous plots in Figures 5.3 because the maximum is expected to gradually evolve towards optimality. The x-axis in Figure 5.4 still numbers generations; again, **Euronet** results in 5.4(a) are presented on a semi-logarithmic scale, with regards to the percentage of extra spare capacity requirements relative to min-costs; and **CselT** plots in 5.4(b) are given in real spare capacity values.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

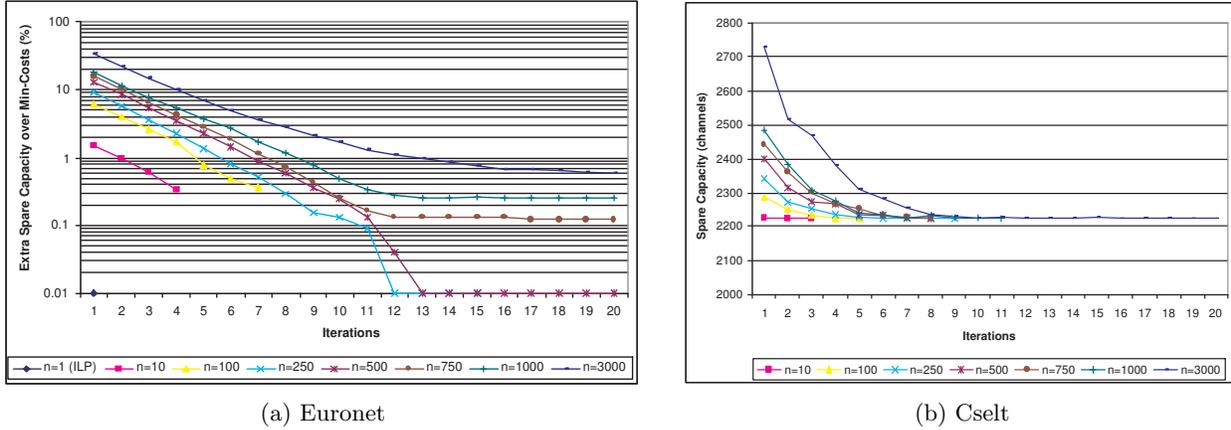


Figure 5.4: *Influence of Population Size on the GA-ILP Performance.*

The GA-ILP always reached convergence in a reasonable number of iterations, typically 20 or less. But the convergence seems slower, in terms of number of generations required, for larger populations with more individuals of smaller sizes than the convergence for smaller populations. In the same manner, **Euronet** instances typically converged in more iterations than **CselT**, for the same number n of individuals per population. Both cases are defensible by the harmful mutation policy applied to small individuals. To illustrate, a population of $n = 3000$ individuals means 28 distinct cycle structures per individual for **Euronet**. Mutation operations typically substitute a random dozen, i.e. almost one half of the 28, for solution-cycles of individuals not ranked for breeding in previous generations. So in order to avoid slower convergences, n should not be too high.

Also in Figure 5.4, varying population size did not affect overall solution quality. For **Euronet** of which an exact solution was obtained by running ILP_1 itself, the GA-ILP based solution was always within 1% of optimality, even for $n = 3000$. Even though it is not possible (for now) to know whether or not GA-ILP solutions are optimal in the case of **CselT**, the GA-ILP always converged to the same objective function value within the MIPGAP being employed.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

5.2.3.3 Checking on the Growth of Final Merged Populations

To check on the growth of final merged populations relative to the size of single individuals, the x-axis in Figures 5.5 recall the number of individuals considered in every population for each specific experiment; and the y-axis gives two histograms indicating the number of distinct simple cycles comprising one single individual for the n under consideration and the number of all unique candidates involved in the final GA-ILP preselection for the same n . A semi-logarithmic scale is used, with x-axis and y-axis crossing each other at 1.

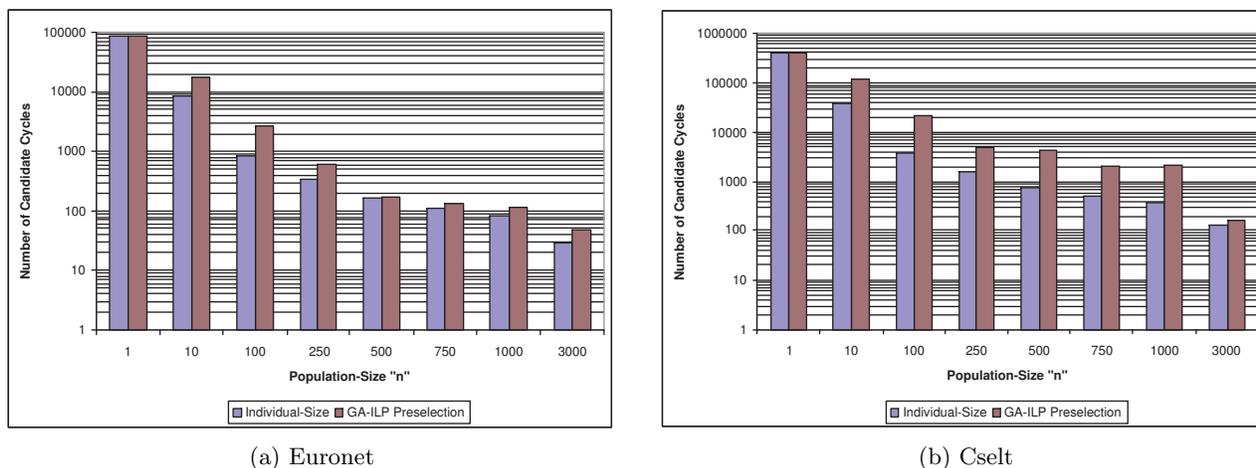


Figure 5.5: *GA-ILP Reduced-Sets of Candidates.*

In both histograms, individuals tend to become nearly identical at the convergence; but smaller population sizes with more distinct simple cycles per individual involve less numbers of iterations and result in last generations of higher relative diversity. For example, the final preselection involves less than twice the number of cycles in both **Euronet** and **Cselt**, for $n = 3000$ that involves the maximum reported number of iterations in both network cases. But this is respectively more than 3 and 5 times for $n = 100$. Nonetheless, the growth on final GA-ILP preselections is still $O(\text{individual sizes})$ for any test case.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

5.2.4 GA-ILP Solution Quality vis-à-vis that of Other Practical Methods for Solving p -Cycle Design Problems

GA-ILP based solutions for **Euronet** and **Cselt** are now compared with what was obtained using prior approaches for solving the p -cycle network design problem. GA-ILP results pertain to a population of $n = 250$ individuals and ILP_1 for conventional p -cycle minimum spare capacity design was considered as the GA constituent ILP. Comparative methods include the two-step approach where feasible (i.e. DFS *plus* ILP_1), CIDA *grow* and CG.

Table 5.1: *The GA-ILP vis-à-vis Prior Approaches: Sample Results for Euronet and Cselt*

Network Instances	DFS & ILP_1	GA-ILP, n=250	CIDA <i>grow</i>	CG
Euronet				
- spare channels	834	834	889, i.e. +6.59%	834, i.e. +0%
- candidates	84,963	611	2237	n/a
- run times	4 min	1h10min	2.41 sec	5min 40sec
Cselt				
- spare channels	Exceeds the	2220	2361, i.e. +6.35%	2224, i.e. +0.18%
- candidates	16GB	4949	3305	n/a
- run times	memory limit	2h30min	3.30 sec	5min 93sec

The comparative Table 5.1 gives test case networks in its first column; two-step based solutions in its second column; GA-ILP preselections and related network solutions in its third column; CIDA network designs in its fourth column; and CG designs in its fifth column. Vis-à-vis DFS & ILP_1 two-steps that provide exact solutions, the GA-ILP reached full optimality for **Euronet**, and achieved a design solution for **Cselt** whereas corresponding instance for DFS & ILP_1 exceeded the 16GB memory limits of the experimental environment. Compared to the others, GA-ILP solutions were of 6% higher standard quality than CIDA-*grow*, and as good as that for CG reported results that are proven to be 1% gap of optimality. This is of special interest and even constitutes a proof of optimality for **Cselt**.

5.2.5 Run Time Issues

If GA-ILP is convincing from a solution quality viewpoint, it remains questionable from run time perspectives. Very high values are recorded in Table 5.1, compared to CIDA and CG completion times. Even for the embryo GA-ILP implementation, it is confusing

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

that a very few iterations are equivalent to so high computational times. The statistics in Table 5.2 may help for a deeper analysis. For every network under study, the first row's entries indicate the number of iterations at the convergence and the corresponding run time, considering various population and individual sizes in the series of experiments. And the second set of rows gives hypothetical time-estimates for the evaluation step alone. Entries respectively pertain to single individuals in a population, all individuals from an entire generation and all iterations in the experiment in question.

Table 5.2: *Deeper Analysis of GA-ILP Run Time Trends, for Euronet and Csel*

Euronet							
n	10	100	250	500	750	1000	3000
$ P /n$	8497	850	340	170	113	85	28
Iterations	4	7	15	>20	>20	>20	>20
Real Duration in (H:min:sec)	01:07:00	00:33:00	01:10:00	02:00:00	03:04:00	04:24:00	43:43:00
<i>Hypothetical times for evaluation step only</i>							
estimate for an individual	00:00:50	00:00:01	00:00:01	00:00:01	00:00:01	00:00:01	00:00:01
estimate for one iteration	00:08:20	00:01:40	00:04:10	00:08:20	00:12:30	00:16:40	00:50:00
estimate for all evaluations	00:33:20	00:11:40	01:02:00	02:46:00	04:10:00	05:53:00	16:40:00
Csel							
n	10	100	250	500	750	1000	3000
$ P /n$	38774	3877	1551	775	517	388	129
Iterations	3	5	9	8	8	11	>20
Real Duration in (H:min:sec)	27:03:00	2:33:00	2:38:00	2:27:00	3:25:00	4:29:00	39:50:00
<i>Hypothetical times for evaluation step only</i>							
estimate for an individual	00:02:00	00:00:20	00:00:10	00:00:01	00:00:01	00:00:01	00:00:01
estimate for one iteration	00:20:00	00:01:20	00:41:40	00:08:20	00:12:30	00:16:40	00:50:00
estimate for all evaluations	01:20:00	00:16:40	06:15:00	01:60:00	01:40:00	03:03:00	16:40:00

Run time entries in Table 5.2 not necessarily increase with the number of iterations required to reach the convergence, especially for extremely small and extremely large populations (e.g. $n = 10$ and $n = 3000$). Considering hypothetical durations for calculating the ILP-based fitness function per individual, it is not the evaluation step that causes computational issues. The truth is AMPL script language is not adapted for implementing GA-like normal steps of encoding, selection, crossover and mutation; we neither know nor control how AMPL manages threads and processes. So to speed-up execution, the GA-ILP implementation requires combining AMPL/CPLEX with such custom programming languages

as C++ or Java. But this first proposal focuses on proving that the GA-ILP works properly and is a promising alternative for solving p -cycle design problems on a very large scale.

5.3 A 200-node Challenge Instance of the Conventional p -Cycle Minimum Spare Capacity Design Problem

The purpose for the GA-ILP is to go on to much larger problem sizes, especially where cycle enumeration is not practical. This case study pursues the goal of solving the conventional p -cycle minimum capacity problem for the 200-node network in Figure 5.6.

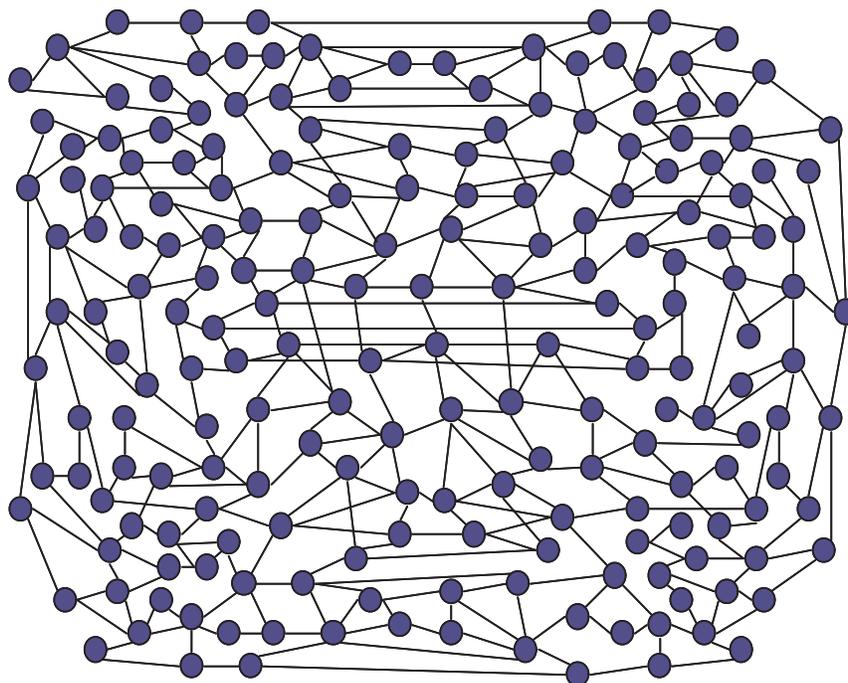


Figure 5.6: *The 200-node Challenge Topology: 200 nodes, 394 spans and a cosmologically large scale of candidate structures.*

5.3.1 The 200-node Test Case Network

The 200-node challenge topology was randomly assembled from copies, foldings, replications and random modification of spans on the well-known **Euronet** transport network.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

The idea behind such variations of a smaller transport network instance is to create a cosmologically large scale problem instance, while retaining the overall characteristics of transport networks in terms of edge locality and nodal degree. The result is as shown in Figure 5.6, and consists of 200 nodes and 394 spans, for an average nodal degree of 3.94. In the series of experiments, the traffic matrix applied consists of 19,900 demand bundles with one demand quantity between every pair of nodes, and demands for instance lightpath quantities uniformly distributed on the interval [0..10]. Minimum-hop routing of the basic working demand flows results in a total of 693,761 working channels to be protected.

With no circumference-size limit, the space of all possible candidate cycles available in the 200-node challenge instance multiplies towards infinity. Based on the rate at which the number of cyclical structures rises for a given graph (i.e. $O(2^{|S|-|N|})$), an estimate of the number of candidates is $O(2^{394-200}) = \sim 2.5 \times 10^{58}$ for the 200-node test case network. Obviously, there is no practical chance to import it into a p -cycle ILP as the resulting problem instance will exceed the maximum number of variables and constraints permissible by the ILP solver. In contrast, the novel combination of GA-methods and ILP seems to have many features to recommend it for such large survivable network design problems. But the GA part of the GA-ILP method only acts as a preselection algorithm, which finds suitably small sets of candidate cycles for a subsequent ILP solution. And the large space of all distinct cycle structures is still separately enumerated for the GA-ILP, by a separate cycle-finding algorithm storing all those cycle structures in a very large disk file. Even under 18 hops alone, the prior-described DFS cycle finder reported up to 15,307,626 distinct simple cycles, which cannot be embodied in an initial population for the original GA-ILP because the result will be too large to remain under server memory limits. The key to extend the GA-ILP method to now cope with essentially unlimited problem sizes is to embed the cycle-finding ability into the GA-ILP itself, as it iterates to develop preselected candidate cycle sets.

5.3.2 Facing Infinite Candidates within the GA-ILP Framework

At this scale, the idea is to no longer enumerate all of the essentially infinite set of distinct simple cycle structures, before selecting a small candidate subset. Instead of ever seeking for the true candidate cycle space, rather incomplete samples are generated by local discovery

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

in the vicinity of each node. The overall philosophy is that it is in the nature of p -cycle problems to comprise many equivalent solutions; so although all candidates are never enumerated, representatively good candidates and collectively good subsets of candidates are very likely to be found just by sampling the full space of all possible cycles by a zeroes of overlapping explorations from each node of the network.

Starting with a node taken at random, the DFS cycle finder is called to enumerate, under a desired hop-limit, a maximum of X cycle structures crossing the random node in question. The enumerated set is then reduced within the original GA-ILP framework, with a surrogate ILP problem instance which identifies a candidate subset that most contributes to a measure of single span failure restorability relative to their own spare capacity. After doing so for every node, the union of the reduced subsets, each resulting from one specific node-based exploration, becomes the final preselected set of candidates for the precise (i.e., non-surrogate) version of the design problem to be solved. Here, that is the p -cycle minimum spare capacity design model with 100% span failure restorability requirement.

The DFS local-cycle enumerator is implemented so as to “virtually” remove nodes following their inspection. The purpose of this node removal is to never consider a given cycle twice from one node inspection to another, thereby avoiding cycle duplications that would increase surrogate subsidiary problem sizes. Adapting the GA-ILP approach in this way, i.e. focus on a single node at once, makes it possible to explore in the space of all cycles as widely as possible; subject to user set time and memory limitations. On the `Cse1t` network instance discussed in Section 5.2, the GA-ILP initial population involved all of the 387,740 cycle structures available within the graph, meaning that each DFS node exploration and its GA-ILP reduction is expected to run on equivalent size of samples and to result in far fewer elite candidate cycle combinations to promote into the final ILP problem.

5.3.3 GA-ILP Subsidiary versus Final p -Cycle Design Models

Because of how the entire cycle space is now locally sampled from each node, it is not possible to use standard ILPs for p -cycle design in each iteration as the result may not be 100% span restorable. The intent of finding collectively efficient “ideas” for subsets of candidates is, however, easily met by defining a suitable subsidiary ILP which only asks for *low unrestorability* and *low spare capacity* for the reduced candidate set identified by each

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

nodal exploration. Here in, ILP₁ for conventional p -cycle minimum spare capacity design is substituted for the surrogate ILP₁₉ in which 100% R_{span} is not strictly required.

ILP₁₉ is given by equations (5.3), (5.4) and (5.5). The bi-criterion objective in equation (5.3) minimizes both unprotected traffic within working capacity channels and spare capacity requirements. With 693,761 working channels in the 200-node case study, the parameter α in equation (5.3) can be set equal to 10^{-6} in order to not upset the principal objective of maximizing restorability while biasing the design towards preselection of cycles that will minimize spare capacity requirements in the final fully detailed ILP. Equation (5.4) implements the measure of unrestorability of the decided set of p -cycles, where unrestorability is the ratio of unprotected over working channels. Equation (5.5) defines the spare capacity of the decided set of p -cycles.

$$\text{Minimize } \sum_{i \in S} (w_i - \theta_i) + \alpha \cdot \sum_{j \in S} C_j \cdot s_j. \quad (5.3)$$

$$\theta_i \leq \sum_{p \in P} x_i^p \cdot \eta^p, \forall i \in S. \quad (5.4)$$

$$s_j = \sum_{p \in P: x_j^p=1} \eta^p, \forall j \in S. \quad (5.5)$$

ILP₁₉ follows prior mathematical symbology for the set of spans $|S|$; input parameters for working capacities to be protected on edges w_i and span costs C_j ; decision variables for spare capacity to be placed on edges s_j and p -cycle copies η^p . In addition,

- θ_i encodes the portion of working channels, on span i , which are protected by the set of p -cycles involved in the design solution. This is an integer less than or equal to the number of working capacity channels on span i (i.e. $\theta_i \leq w_i$).
- Here in the GA-ILP with embedded DFS cycle enumerator, P is the set of cycles found from one of the node explorations. But again P may be populated for different purposes with either all candidate cycles as within the original GA-ILP framework, or with candidates comprising the individual under evaluation, or with cycles matching a single DFS node inspection, or at the completion of the overall GA-ILP with the preselected candidate set. Thus without changing p -cycle ILP models themselves, P may be populated

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

for different purposes with either all candidate cycles, or within the GA-ILP framework with candidates comprising an individual, or at the completion of the GA-ILP with the preselected candidate set.

5.3.4 Detailed Analysis of the DFS Local-Cycle Enumerator

The key to cope with infinite candidate spaces is to enumerate and reduce candidate cycle subsets on a per-node basis. This series of experiments considered network nodes in turn and in an increasing order—i.e. $N_1, N_2, N_3, \dots, N_{198}, N_{199}, N_{200}$. In each node exploration, the DFS local-cycle enumerator was called to find all cycle structures containing the node in question within a circumference limit of 18 hops, as protection structures larger than this are probably not desirable in practice; the enumerated candidate subset was then reduced within the GA-ILP framework and the inspected node plus its incident spans were “virtually” removed from the network graph before going with the next node exploration.

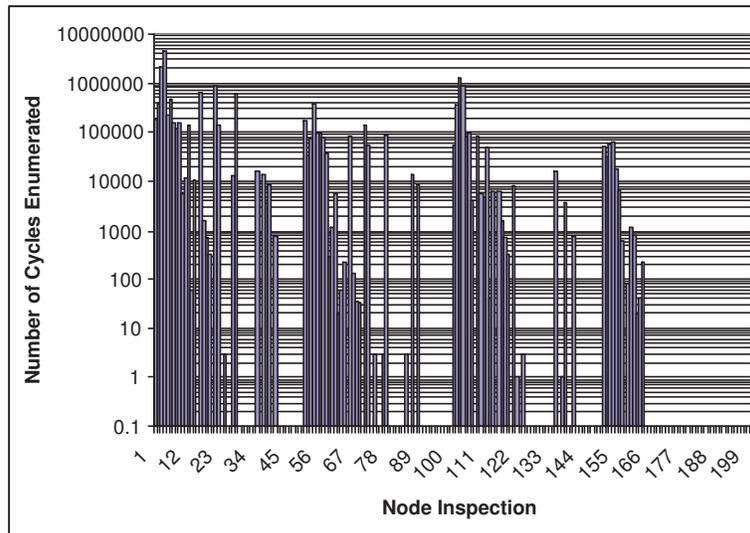


Figure 5.7: *DFS Local-Cycle Enumeration*

As a starting point, Figure 5.7 gives the number of cyclical structures per node exploration in a semi-log scale: the x-axis identifies the node under inspection and the y-axis counts cyclical structures crossing the node in question. In interpreting Figure 5.7, it is noticeable that up to 8,316,139, meaning 54% of the total space of 15,307,626, candidate

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

structures were found during the ten primary node-explorations alone. And with subsets of 10,824,365 and 12,182,470 candidate cycles resulting from N_{50} and N_{100} node inspections, the remaining portion of the graph which is still 75% and 50% of the original network contains less than 29% and 20% candidate structures. This means if the DFS local enumeration confines cycles under certain circumference limits, or candidate subset cardinality to a certain number, this will restrict circumference-size for earlier node inspections alone, that are the ones returning huge candidate cycle subsets.

Sometimes in Figure 5.7, earlier cycle enumerations entirely cover a specific node. Inspecting the node in question will consequently provide no or very few candidates—e.g. Figure 5.7 reports as much as 54.5% (with 109 over 200) of the node explorations ending up with zero candidate structures. This observation suggests that inspecting every node may not be required in general but it will be seen that running the GA-ILP with an integrated DFS local-cycle enumerator always provides a final preselected set of candidate cycles that does cover all of the network areas. Another implication of the previous observation is that in cases where a node exploration results in very few candidates, the GA-ILP reduction is not applied but the full enumerated-set is instead preselected.

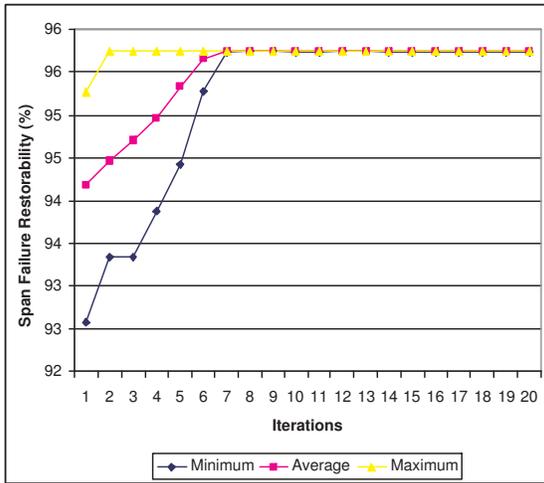
5.3.5 GA-ILP Convergence under a Surrogate p -Cycle ILP

In this series of experiments, a maximum of 100,000 cycle structures is required per node exploration. Each pre-enumerated set is then reduced within the GA-ILP framework, with initial partition sizes guaranteeing no more than one thousand preselected cycles in merging final-stage individuals for the node-based exploration in question. There will be 200 individuals of 500 cycles each for enumerated subsets of 100,000 candidates, as Section 5.2 indicated a final growth that is less than twice the size of such an individual within the GA-ILP framework. The size of the initial population is adjusted for candidate subsets of lower cardinality; for example, the initial population will consist of 24 individuals of 500 or 501 candidates each for the node exploration N_{10} that resulted in 12,004 enumerated cycle structures.

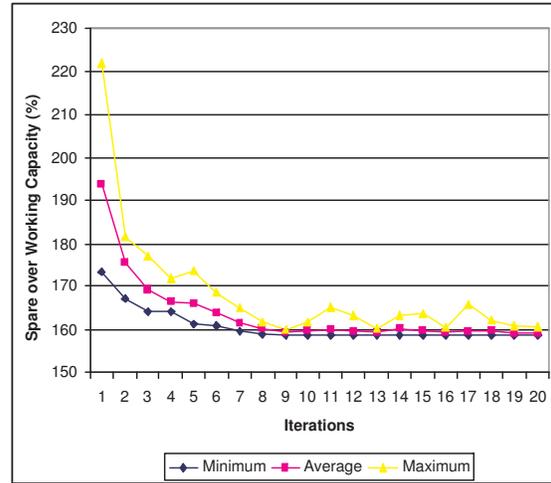
5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

5.3.5.1 Convergence of One Single Node Exploration

Figure 5.8 shows details on the GA-ILP convergence during the fourth node exploration. The x-axis numbers iterations while the y-axis relates to two different metrics: span failure restorability in 5.8(a) and corresponding spare capacity requirements in 5.8(b). For each of the Figures in 5.8, three plots indicate maximum, average and minimum values for individuals of the current generation. Interpreting the results, the GA-ILP algorithm still converges but has some difficulties to reach a steady state because of the bicriteria surrogate ILP. More specifically, the convergence occurs by the 7th iteration in 5.8(a) and by the 9th iteration in 5.8(b); but contrary to the original version of the GA-ILP, convergence spare capacity values fluctuate when the algorithm continues to run until the 20th iteration.



(a) Maximization of $R_{1-\text{span}}$



(b) Minimization of Spare Capacity Requirements

Figure 5.8: GA-ILP Convergence for the Fourth Node Inspection.

5.3.5.2 Convergence of the Overall Process

The final GA-ILP preselection consisted of a collection of 21,720 high-merit candidate structures. Figure 5.9 provides details on the effectiveness of this preselected set. The x-axis lists nodal explorations and the y-axis shows corresponding single span failure restorabilities and equivalent spare capacity redundancies. Both metrics were obtained by solving the subsidiary version of the conventional p -cycle network design problem considering as

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

candidate-set the union of GA up-to-date reductions—i.e., while merging preselected subsets until the current node expedition. The blue plot in Figure 5.9 shows that $R_{1\text{-span}}$ restorability level improves over DFS *plus* GA-reductions, and reaches 100% by the 103rd node inspection. Within the prior-defined ILP symbology, $R_{1\text{-span}} = 1 - \frac{\sum_{i \in S} (w_i - \theta_i)}{\sum_{i \in S} w_i}$ calculates single span failure restorability level for span i .

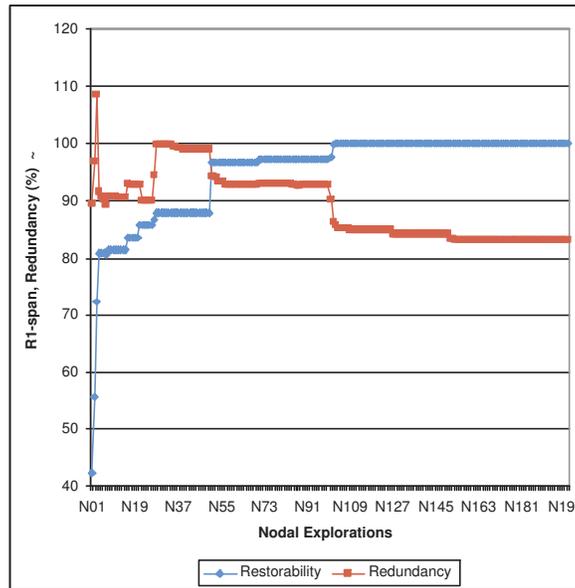


Figure 5.9: *Development of the Fully Span Restorable 200-Node Network Design.*

Regarding spare capacity requirements, improvements become effective after full span restoration is achieved. The final solution in Table 5.3 consists of 195 p -cycles with a total of 41,865 unit-sized copies that are built in 576,853 spare capacity units, for 83% redundancy. Based primarily on the convergence behavior and the fact that the final design assembly is solved to full termination by ILP₁, it appears that the result is optimal; but one cannot strictly claim that. To our knowledge, this is the first reported attempt to design such a large scale p -cycle network. The topology and demand data have been made available to anyone working on very large scale solution methods, which are able to cross-check the results here. Interestingly, Montréal colleagues working on the CG-based approach told us they could not solve the 200-node problem instance; they experienced memory limits. Therefore, it is difficult to know, by cross-checking with CG as hoped,

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

whether 83% redundancy is actually the optimal or a near-optimal result for this network or not.

Table 5.3: *Characteristics of the 200-Node p -Cycle Network Design Solution*

SuperGraph	Working Capacity (channels)	Spare Capacity (channels)	Redundancy (%)	Distinct Structures	p -Cycles
<ul style="list-style-type: none"> - 200 nodes, - 394 spans, - 3.94 nod. deg. - Infinite set of candidate cycles 	693,761	576,853	83	195	41,865

5.3.6 Run Times

Regarding run times, the aim of the GA-ILP proposal is to first be able to get efficient solutions on very large scale problems, as opposed to having no possibility of a solution at all. Speed-related optimization of the steps for the GA-ILP method is not even attempted yet. Nonetheless, the nodal cycle subset enumeration of candidate structures was the most time consuming step of the GA-ILP process, with almost 3.5 hours in total to visit and explore from all of the network nodes. Each node's cycle enumeration was followed by a size-reduction process essentially based on a subsidiary ILP and GA-like steps, both completing in a few minutes (per node) because they never faced individuals of more than a thousand candidates. After merging all reduced sets to build the final space of candidates for a complete final p -cycle ILP, the solution was obtained in about 30 minutes. Thus, the total computation time is estimated to be about 5 hours. This is just an estimate because at present, the succession of steps and files used are manipulated manually but could be put in an automated script. Five hours is estimated if the steps were fully automated.

5.4 p -Cycle Design Problems with Many Complicated Practical Constraints

In a more general manner, the GA-ILP is expected to solve p -cycle problems with many complicating practical constraints such as joint working and spare capacity optimization, wavelength assignment, transparent reach limitation in restored network states, modular

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

capacity, node restorability maximization, multi-QoP, availability-managed p -cycle networks and dual failure restorability enhanced.

5.4.1 Truth about Sample Results in Previous Chapters

Even though the selection of test case networks in Chapters 3 and 4 was essentially of small and medium sizes, many of them actually involved huge numbers of variables and constraints, slow run times and other computational issues due to the complexity of p -cycle advanced considerations under exploration. Table 5.4 records the total number of variables and constraints involved in p -cycle problem instances within the thesis. Gray cells identify cases where the GA-ILP was required to provide design solutions.

Noticeably, ILPs for hybrid and fully transparent wavelength switched p -cycles and glass switched p -cycles remained solvable, despite the huge number of variables and constraints for most network instances. However, an observation was made that the computation is fairly slow, compared to regular size model instances ILP_{1, 2; 7; 14} and 15. This suggests that the complexity not only depends on variables and constraints, but also on the form of the problem and possible solutions. (Data in Table 5.4 differ from the complexity reported in Table 4.3, which did not considered reach limitations.)

The GA-ILP was typically required where larger candidate cycle spaces were coupled with complex p -cycle problems with many complicated constraints. Those network case instances essentially include `Cost239` and `Euro`, and sometimes the `Bellcore` problem instances of ILP_{3 to 6} pertaining to two-hop, flow or `nepc` for R_{1-node} maximization; all ILP_{8 to 13} for general path-protecting p -cycles; and ILP_{16 and 17} for optical reach limit in restored network states.

5.4.2 Effectiveness of the GA-ILP for Solving Advanced p -Cycle Network Design Problems

5.4.2.1 Convergence and Performance

Considering any large scale p -cycle problem, an appropriate variant of the GA-ILP can be derived by just substituting the GA constituent ILP for the p -cycle mathematical design model under consideration. The algorithm is expected to converge through optimality as

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

Table 5.4: Complexity of Advanced p -Cycle ILPs for Several Network Instances

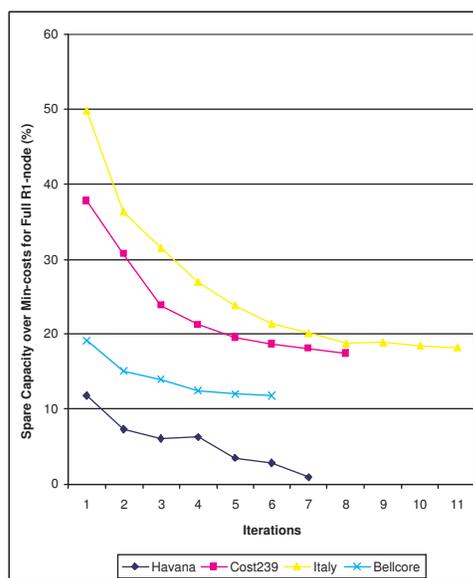
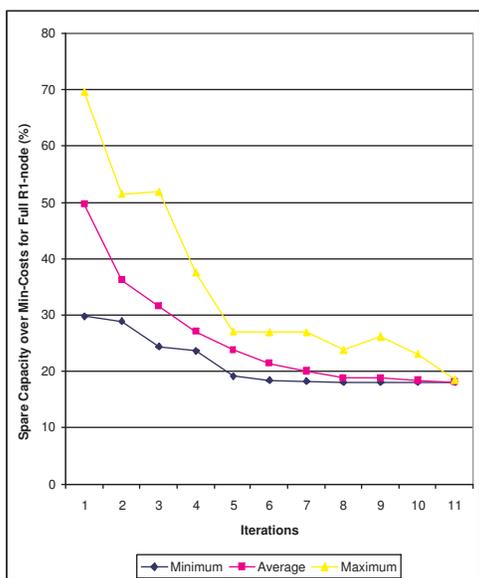
Models	Problem Description	Havana	Cost239	Italy	Bellcore	Euro
ILP ₁	reference p -cycle min spare capa design model.	3,776	95,468	14,046	28,445	30,268
ILP ₂ (2-hop & flow)	max R_{1-node} , given a 100% span restorable design.	30,069	31,376	46,398	63,874	877,788
ILP _{3 to 5} (2-hop & flow)	R_{1-node} under min spare capa; R_{1-node} with controlled penalties; full R_{1-node} and R_{1-span} .	678,952	10,858,100*	2,857,880	7,751,980	36,357,700
ILP ₂ (nepc)	max R_{1-node} , given a 100% span restorable design.	25,691	120,420	49,983	91,975	720,533
ILP _{3, 4 and 6} (nepc)	R_{1-node} under min spare capa; R_{1-node} with controlled penalties; max possible R_{1-node} .	345,021	6,587,480*	1,898,780	5,823,070	24,135,400
ILP ₇	min spare capa design for two-hop segment protection	202,018	5,134,010	701,092	1,648,990	2,628,540
ILP ₈	min spare capa design for general path-protecting p -cycles (GPP).	28,269	616,932*	124,879	204,606*	561,939
ILP ₉	imposing the failure independence constraint to an existing GPP design.	27,790	529,549	130,914	221,513	777
ILP ₁₀	similar to ILP ₂ , but for GPP.	96,074	607,181	291,588	327,617	833
ILP _{11 to 13}	similar to ILP _{3 to 5} , but for GPP.	966,669	15,649,700*	4,095,030	7,102,120*	52,945,500
ILP ₁₄	min average restored state path length model, given a 100% span restorable design as input.	55,715	181,675	112,279	163,000	1,392,410
ILP ₁₅	min longest restored state path, given a 100% span restorable design as input.	100,957	258,897	202,137	286,482	2,698,780
ILP _{16 and 17}	min average length and minimax under min spare capa; length constrained under a max reach limit; shortest possible minimax.	2,548,650	63,261,300*	12,878,400*	35,288,200*	115,495,000
ILP ₁₈	min spare capa design for glass switched p -cycles.	2,553,220	63,265,400*	12,883,700*	22,829,400*	115,522,000*
ILP ₁₉	surrugate p -cycle model for use in cases where the GA-ILP faces infinite candidate sets.	N/A, no specific sample results in the thesis.				
Other ILPs Appendix B	min spare capa for opaque p -cycle configurations.	6,824	97,898	17,372	31,871	56,364
	min spare capa for hybrid p -cycle configurations.	2,558,730	63,410,100*	12,906,500*	22,869,400*	115,550,000*
	min spare capa for fully transparent wavelength switched p -cycles.	2,559,850	63,411,200*	12,907,600*	22,870,600*	115,552,000*

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

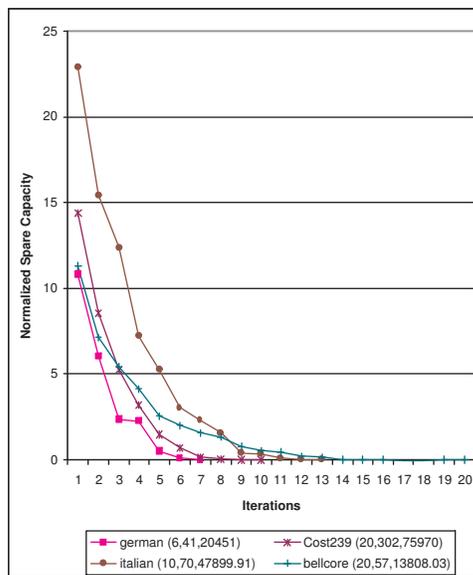
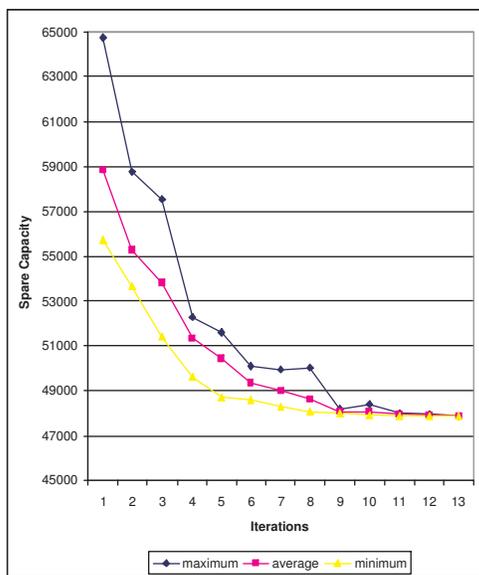
before, i.e. in a few iterations and with no influence of population sizes. To illustrate, Figure 5.10 shows the GA-ILP_{5 and 8} convergence for full protection against both node and span failures, and for general path-protecting p -cycles (GPP). In the primary set of Figures 5.10(a)-(b), the x-axis numbers iterations and the y-axis indicates spare capacity requirements for full R_{1-node} versus min-cost requirements for full R_{1-span} only. Considering a population of $n = 20$ individuals, 5.10(a) gives details on the GA-ILP₅ convergence for **Italy** network by plotting maximum, average and minimum extra spare capacities for individuals of each given generation; while 5.10(b) focuses on the average but for all test case networks considered for experiments, with population sizes of $n = 6, 40, 20, 20, 20$ for **Havana**, **Cost239**, **Italy**, **Bellcore** and **Euro** respectively. The second set of figures also pertains to two series of experiments. In 5.10(c), the GA-ILP₈ convergence is given for the **Italy** network only, considering a population size of $n = 10$. The x-axis numbers iterations and the y-axis returns maximum, average and minimum spare capacities required to provide 100% span restorable design solutions, using in turn individuals of the given iterations as candidate sets. The x-axis in 5.10(d) also numbers iterations while the y-axis shows the convergence of spare capacity requirements over iterations for all test case networks (except the **Euro** case, to be discussed later). Population sizes are respectively of $n = 6, 20, 10, 20$ for **Havana**, **Cost239**, **Italy** and **Bellcore**; and to accommodate all plots in 5.10(d), spare capacities are normalized on the basis of the value at the convergence.

Interpreting **Italy** related plots in Figures 5.10(a) and 5.10(c), all of the three metrics (i.e. maximum, average and minimum objective function values) become equal after a few iterations. This means the GA-ILP algorithm converges for more complex constituent p -cycle ILPs, as was the case with ILP₁ for conventional minimum spare capacity design. Regarding the series 5.10(b) and 5.10(d), the GA-ILP reached optimality at the convergence for **Havana** and **Italy** network cases of which corresponding instances for ILP_{5 and 8} were also directly solvable using the two-step approach of cycle pre-enumeration and ILP optimization. So even though there is no benchmark of comparison for **Cost239**, **Bellcore** and **Euro**, GA-ILP network design solutions can be trusted based on the algorithm convergence which was proven optimal for **Havana** and **Italy**.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales



(a) GA-ILP₅ Convergence for Italy Network Case (b) GA-ILP₅ Convergence and Solutions for All Networks



(c) GA-ILP₈ Convergence for Italy Network Case (d) GA-ILP₈ Convergence and Solutions (n, presel, sol)

Figure 5.10: GA-ILP₅ and 8 for full R_{1-node} and General Path-Protecting p -Cycles.

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

5.4.2.2 Influence of Population Sizes

To assess the influence of population sizes on the GA-ILP convergence for more complex p -cycles, a series of experiments was conducted by varying population sizes n in the **Italy** network. Table 5.5 reports results: each column in the first row corresponds to a specific population size; the second row gives corresponding individual sizes; and the third and fourth rows indicate the number of generations and spare capacity requirements by the convergence; the last row entry calculates the gap of optimality of spare capacity values at the convergence. Populations of one or two individuals characterize the direct usage of the two-step strategy consisting of candidate pre-enumeration followed by ILP₈ completion.

Table 5.5: *Running the GA-ILP₈ with Different Population Sizes for the Italy Network*

n	1 (ILP ₈)	2	4	6	8	10	20	30	40	50
P /n	557	279	140	93	70	56	28	19	14	12
iteration	n/a	n/a	12	14	15	13	21	27	>30	>30
spare capa.	47,735	47,735	47,837	47,864	48,023	47,900	48,962	50,495	50,535	53,666
Δ (in %)	n/a	n/a	0.21	0.27	0.60	0.34	2.57	5.78	5.89	12.42

With data recorded in Table 5.5, the GA-ILP typically converges in fifteen or less iterations and within 1% of optimality, especially for population sizes of 4 to 10 individuals of more than fifty candidate cycles each. But in discussing GA-ILP₁ for conventional minimum spare capacity design, very large populations with individuals of less than fifty candidate cycles were unappropriated because of a dozen (random) substitutions during the mutation process. Specifically, before the algorithm strictly converges in such medium size networks as **Italy**, very high population sizes of 20 or more result in many iterations and in a deterioration of the solution quality standard; this is because individuals now comprise less than 30 candidate structures (versus a dozen deleterious substitutions), due to the initial candidate space no longer involving huge numbers of cycle structures.

5.4.2.3 Run Times

As GA-ILP applications to advanced p -cycle problems are typically conducted on medium size networks, because of the complexity of ILPs under considerations, quite satisfactory run times of about 5 to 15 minutes were observed. This is because neither many individuals n nor large individual sizes $|P|/n$ are manipulated in the GA-like normal steps. Of course this might be lowered to less than or a few seconds with a more customary implementation; but for now the GA-ILP provides high standard solution quality whereas other CIDA, CG and enumeration-free ILPs are too difficult to be extended.

5.4.3 Multi-Criteria Optimization

In one sense, the GA-ILP did not always converge when considering bicriteria optimization models. The plots for maximum, minimum and average objective function values required to be equal at the convergence did not intersect when using ILP_{3, 6, 16} or 19 as the GA constituent ILP. But it was also noted that minimum, maximum and average values still reached a steady state separately, no longer changing after a few iterations. Terminating conditions were modified accordingly; as a result, both criteria in the objective function respected convergence requirements separately (e.g. Figure 5.8 for ILP₁₉). But using GA-ILP preselections as candidate sets for final fully detailed ILPs, network design solutions still appear to be within 1% of optimality on instances of which exact solutions are known.

In the same manner, solving GA-ILP_{8–13} instances of GPP-related problems for the Euro network showed many subproblem instances resulting in infeasibility by the second iteration. Working on slightly larger individuals did not overcome that issue, but added too much complexity to the ILP fitness function instead, because all individuals of the population successfully completed in the very first iteration. Going back to network characteristics, Euro shows the particularity of many more demand flows than any other test case network, which is key in path-protecting p -cycle problems. This brought up an idea that the GA-ILP should consider as many constituent ILPs as the number of criteria in the objective function for the problem under consideration—e.g. one process to build disjoint-route sets (DRS) and a parallel preselection of candidate cycles!?

5.5 Concluding Points and Possible Future Directions

This chapter introduced a novel combination of genetic algorithm methods and integer linear programming, referred to as GA-ILP, where a GA is guided by an ILP to preselect candidate cycles for a size-reduced final ILP. To face infinite candidate spaces for very large problem instances, a variant of the basic GA-ILP using a relaxed or subsidiary p -cycle model and never-fully-enumerating candidate cycles was detailed as well. Overall, the GA-ILP was interpreted as a generalized preselection of a reduced number of “highly promising” candidate cycles, but with the addition of GA-like attributes wherein a subsidiary ILP is used as the fitness function to “breed” a collection of high merit candidate cycles (and importantly, combinations of cycles) to present a final fully detailed design model solved with a reduced space of candidate cycles. The GA-ILP also showed some conceptual analogies with CG-like aspects in that GA-like normal steps find and retain *good ideas* for the final ILP while CG-dual finds new important constraints for a primal ILP.

Experiments conducted followed the identification of different classes of problems based on whether the candidate cycle space is enumerable and solvable, enumerable but impractical to import into an ILP, or not even enumerable. One series of experiments specifically sought the goal of achieving near-optimal solution for p -cycle network design problems involving 200 nodes; at which problem size the space of all candidate cycles of the graph could not even be enumerated in practice, let alone set into an ILP problem instance. The problem of the creation of a 200-node test case network of controlled properties was discussed as well, in order to later estimate how near to optimal the solution is. And in a third series of experiments, the GA-ILP was applied to advanced p -cycle design problems with many practical constraints, where high solution quality was desired but proof of optimality is not necessarily required.

As a result, GA-ILP solutions were typically found to be within 1% of optimality where the strict ILP approach and/or CG methods provided reference solutions. Moreover, the GA-ILP proposed a 200-node network design solution whereas no other practical method (to date) could offer the capability. Also significantly, the GA-ILP appeared as an entirely repeatable meta-heuristic that simply re-uses existing p -cycle ILPs. This stands in contrast to preselection methods and CG, which would have required subtlety to define effective preselection criteria or reduced costs, and enumeration-free related ILPs that would have

5. A GA-ILP Heuristic for p -Cycle Design on Very Large Scales

increased in complexity. In addition to p -cycle problems, the GA-ILP philosophy is fully extendable to any other field and large scale ILP problems requiring the preselection of highly promising candidates, from a fully known but too large solution space.

However, the GA-ILP was found to be limited in two main aspects. Although the GA-ILP appeared a priori as a meta-procedure of which implementation requires no custom programming, it was found that using AMPL/CPLEX for ILP fitness evaluation in combination with C++ or Java programming languages for GA-like normal steps, will speed-up the overall process. From multi-criteria optimization perspectives, a new variant following the multicriteria-GA framework appears necessary in order to deal with ILP problem instances involving many demand bundles, large candidate spaces and/or bi-criteria objectives.

Chapter 6

Conclusion and Further Work

The following summarizes our contributions, highlights innovations and implications, outlines related publications, and indicates possible directions for future investigations.

6.1 Summary of the Contribution

The research topics considered in this dissertation pertain to node failure recovery maximization, optical reach control and wavelength continuity requirements for transparent-based designs, and network optimization on a very large scale.

6.1.1 Node-Protecting p -Cycles

In developing the topic of node failure protection, the literature review pointed out four practical methods that may lead to node-protecting p -cycles: (i) the p -cycles' inherent ability to recover on-cycle flows transiting through a failed node that was not found to be sufficient to achieve high node failure recovery levels at low capacity costs; (ii) node-encircling p -cycles (NEPCs) which, in theory, would be simple to design and operate if they did not require the use of non-simple cycles and unreasonable capacity requirements in order to provide the node failure protection capability; (iii) flow-protecting p -cycles which are really capacity efficient but require a centralized management to localize a failure and activate the right restoration process; and (iv) failure-independent path-protecting (FIPP) p -cycles which are especially complex to design.

The node-encircling constraint, imposed on NEPCs, was identified as the principal cause of the scarcity of NEPC structures and capacity-inefficiency incurred. And the complexity of proper backup activations was associated with the undetermined nature of flow-protecting p -cycles, structured indefinitely long. The parallel relaxation of node-

encircling requirements for NEPCs and the shortening of failed flows to exactly two-hops gave rise to a simple generalization of how nodes in a BLSR ring or span-protecting p -cycles (to date) derive survivability at the nearest two-neighbors on the same ring. In the experiments, this new insight and approach to node failure recovery, using one single set of ordinary span-protecting p -cycles, provided the intended capability in terms of simplicity, straightforward failure detection and backup activation, and cost-effectiveness.

As the “two-hop flow” strategy for node failure recovery with ordinary span-protecting p -cycles appeared to be a good compromise between all related concepts, the principle was amended in order to protect *two-hop segments* consisting of two adjacent spans plus their common node. The two-hop segment protection paradigm was differentiated from the prior two-hop flow view of ordinary p -cycles, which required no specific arrangements whereas the former transformed any given graph instance into a set of nodes and two-hop segments (as opposed to nodes interconnected by spans), and then performed the routing of demands in the two-hop segment graph. One of the main advantages of the new two-hop segment protection paradigm was to no longer distinguish between span and node failures in the design. And in the experiments, typically, two-hop segment protecting p -cycles gave rise to network solutions requiring lower capacity than what would be required with span- and path-protecting p -cycle architectures.

For comparison purposes, FIPP p -cycle design methods were revisited as well. ILP, DRS and CG design approaches available within the literature were not retained because of such reasons as computational issues, network scalability or overall complexity. Instead, general path-protecting p -cycles (GPPs) were introduced, with the aim of relaxing the constraint of failure independence which greatly complicates, in our opinion, FIPP modeling problems. Then, FIPP solutions were obtained by solving the GPP problem, re-imposing the constraint of failure independence to GPP solutions in order to identify unprotected working paths, and duplicating shorter p -cycles in a way that achieves the goal of full restorability. Interestingly, the experiments revealed less than three working paths remaining unprotected when GPP designs were characterized from the failure independence perspective. As a result, FIPPs through GPPs were much simpler than basic FIPP ILPs, much more efficient than FIPPs through DRS, and more accessible than FIPPs through CG from an implementation perspective. On the other hand, FIPP-GPP allowed the mea-

surement of the effect of relaxing the disjoint route set constraint, as is done with FIPP-ILP and FIPP-DRS approaches; as well, GPP gave rise to a true comparison between FIPP and span-protecting p -cycles; both studies were not possible before because of the lack of trustworthy and accessible FIPP results.

6.1.2 CapEx Concerns and Transparency Fundamentals

Another study in the thesis' series sought to verify whether or not planning networks on the basis of capacity requirements incurred in the solution correlates with real-world CapEx costs. In addressing this economical concern, the NOBEL standardized cost model was adapted to span-protecting p -cycle architectures, as it normalizes CapEx costs for all relevant pieces of equipment building a consistent WDM layer. NOBEL cost derivations distinguished between p -cycle implementations in opaque, hybrid and fully transparent network contexts. A preliminary review of those p -cycle configuration types highlighted the simplicity of opaque p -cycle designs vis-à-vis hybrids and fully transparent p -cycles, which require wavelength assignment and wavelength continuity constraints. But the side effect was the requirement of wavelength conversion capabilities at every node across the network and/or at the p -cycle access points, which significantly increased opaque and hybrid CapEx costs. Although wavelength conversion is of no consideration in fully transparent p -cycles, they surprisingly remained quite expensive. The fact is, working lightpath channels and p -cycle protection channels were no longer allowed to ride onto the same fibers, resulting in a proliferation of fiber optics across the network.

Recognizing wavelength-selective switching as of primary considerations in fully transparent networks, we made a first proposal for the whole fiber switched p -cycle alternative. In comparison with wavelength switched p -cycle architectures, the design complexity was almost equivalent to that of opaque p -cycles, which is much simpler than that of hybrids and fully transparent p -cycles. Despite fully transparent considerations, the complexity decreased because whole fiber switched p -cycles structurally restore entire DWDM wavebands with no switching or manipulation of individual lightpaths, such that the DWDM layer would never know the break happened. Based on actual state-of-the-art whole fiber cross-connect switches, the idea of glass switched p -cycles was proven ideal from a technological viewpoint. And regarding CapEx costs, as long as the loss budgets were adequate,

environments in which fiber switching devices were low cost and ducts were full of dark fibers provided a very low cost alternative to protect an entire DWDM transport layer (or working capacity envelope) against the single largest cause of outage.

When comparing wavelength and glass switched p -cycle architectures from the optical reach perspective, we found a need for controlling path lengths in hybrid and fully transparent wavelength and glass switched p -cycles, contrary to opaque contexts where a wavelength conversion was performed at each node traversed by any path either in normal or restored network states. In the literature, approaches to control restored state path lengths limited the maximum allowable circumference for candidate cycles, or the longest protection path-segments through candidate cycles, considered in the design. Such approaches were found well-suited for hybrids; however, the upper limit to candidate circumference sizes shortened the diameter of the transparent domain in fully transparent wavelength and glass switched p -cycle contexts. The reason is, the upper limit in question was set global to all working path lengths, so that any length was adequate under restored network states. On the contrary, we defined a simple matching principle with direct considerations of single end-to-end path lengths in the restored state design. This new principle consisted of a complementary matching of longer working paths with shorter protection path-segments through available p -cycles, and vice-versa. With it, both the length of protection segments and non-failed portions of any affected working path were taken into account in the survivable design.

6.1.3 p -Cycle Design on a Very Large Scale

Within the thesis, each new insight and principle proposed was typically built into four ILP mathematical design models. One of them was aimed at characterizing an existing 100% span restorable design from the new insight perspective. Two other ILPs sought to offer the new insight capability at best-effort, with no penalty or under given maximum extra spare capacity budgets over min-cost requirements for 100% span failure restorability alone. And the fourth ILP model provided the insight capability in question at a full satisfaction level; in doing so, capacity penalties over min-costs were possibly incurred. However, many of those p -cycle mathematical design models were unsolvable for medium size network instances, due to many candidate cycles involving a huge number of variables and complicated practical constraints.

Ultimately, p -cycle problems were differentiated on the basis of whether the set of all possible candidate cycles was fully enumerable and solvable, fully enumerable but impractical to import into the ILP solver, or not even enumerable in practice. Solving instances of the second and third problem classes required the use of advanced p -cycle design methods, capable of addressing large scale problems. Reviewing such methods within the literature, preselection approaches were found of great conceptual simplicity, but they led to sub-optimal solutions only because they are choosing candidate cycles on the basis of their individual merit. On the other hand, enumeration-free ILPs and CG heuristics gave rise to optimal or near-optimal solutions but providing such a capability was time consuming and involved too much complexity. Importantly, all of the preselection, enumeration-free and CG methods essentially focused on span failure restorability purposes; they were not easily amenable to more complicated p -cycle problems.

Our approach to manage the size of large scale instances related to p -cycle preselection techniques. But rather than choosing candidates on the basis of their individual merit, a GA-like evolutionary heuristic was guided by a p -cycle ILP model to preselect a combination of *collectively high merit* candidate cycles, in order to populate a size-reduced final ILP. As other preselection methods, the novel GA-ILP has the attraction of greater conceptual simplicity which makes it accessible to the average user. With the ILP-integrated philosophy, the GA-ILP heuristic is entirely repeatable with any existing p -cycle ILP. And in the experiments, the GA-ILP was successfully applied to very large scale p -cycle problem instances, including a 200-node challenge case.

Even though test case networks were of almost any size, the ILP solver never faced impracticably large sub-problem instances because GA-like attributes allowed the control of individual sizes. Rather, because the GA constituent ILP captured the exact goals of any p -cycle problem under consideration, the GA-ILP typically provided exact or at least approximate solutions, at the same quality standard as enumeration-free ILPs and CG heuristic. Importantly, the GA-ILP provided the world's largest instance of a p -cycle network design problem; it solved a 200-node network instance of the conventional minimum spare capacity design problem whereas no other practical method could offer the capability. Moreover, many of this thesis' tests required the use of the GA-ILP to solve newly formulated p -cycle problem instances on medium-size networks.

6.2 Innovative Aspects of the Thesis and Implications

Thus, many new insights and approaches were developed in this dissertation. The GA-ILP is probably the major contribution. To date, the literature shows no use of an ILP mathematical model as the fitness function for a GA-heuristic as was done with the GA-ILP. The principal motivation for this line of reasoning is to no longer tweak appropriate fitness functions for GA-based problems. Instead, the GA-ILP just re-uses existing ILP mathematical models, which capture the exact goals of any specific problem statement. And the very same GA-ILP heuristic developed for a given problem is entirely repeatable for a wide range of problems. Considering different and/or more complex aspects of the initial problem statement, just the GA constituent ILP is actually substituted for a new mathematical model reflecting the advanced considerations. Additionally, the GA-ILP exclusive combination of genetic algorithms and integer linear programming tackles very complicated problems on quite large scales and provides solutions within 1% of optimality. Especially here in the thesis, complex p -cycle models with many complicated practical constraints were successfully solved within the GA-ILP framework. Moreover, the GA-ILP provided a 200-node network design solution whereas such well-established methods as CG could not offer the capability. This 200-node challenge case is the world's largest solved instance of the conventional p -cycle minimum spare capacity design model, and p -cycle problems in general.

Another advancement of this research is the two-hop segment protection paradigm as opposed to span- and path-protecting schemes, or the flow-protecting option specific to p -cycles. Two-hop segment protection is subtly different from flow protection as the former does not operate on (parts of) demand working routes, but on the network topology itself. This two-hop segment protection principle retains the simple and local type of failure detection and switching reaction for span-protecting schemes. Moreover, instead of considering span failure conditions alone, the protection lightpath for a given two-hop segment responds if one of the adjacent spans forming the segment in question or their common node fails. Certainly, two-hop segments sharing protection channels on parts of their restored state paths (here, the same p -cycle) must be mutually disjointed from each other. But relative to transport-level demands, there are very few two-hop segments

across a network. So checking on the rivalry of two-hop segments is quite an easy task in comparison with the computational complexity incurred with path-protecting paradigms.

As well, this thesis explains and explores two p -cycle insights and principles that stand in contrast to the widespread idea that span-protecting paradigms can neither respond to node failure events nor allow the control of the diameter of transparent domains. On one hand, prior research in the literature overlooked the intrinsic ability of span-protecting p -cycles to recover any two-hop flow transiting through a failed node when the two spans adjacent to the failure node both end on other nodes of the same p -cycle. In fact, this simple generalized approach to node failure recovery with span-protecting p -cycles extends the BLSR-like loopback reaction to recover on-cycle flows transiting through a failed node on the ring. Those intrinsic node-protecting capabilities of p -cycles were derived in the same way as the BLSR-loopback behavior under on-cycle span failure conditions was extended and combined to a breaking reaction in the event of straddling span failure, in the advent of ordinary p -cycles. On the other hand, this thesis introduces a complementary matching of longer working paths with shorter protection segments through available p -cycles, and vice versa. Although ordinary p -cycles are primarily intended to act locally, at the end-nodes of a failed span, the simple matching principle elegantly controls *end-to-end* optical path lengths in the restored network state, as related designs now take into account both the protection segments and the non-failed portions of any affected path for every potential failure scenario.

Unlike most network survivability schemes that assume protection at the wavelength-switched granularity level, the p -cycle literature briefly mentioned the configurability of p -cycles at the fiber-level of granularity. This dissertation seriously considers the topic and makes a first proposal using whole fiber switched p -cycles for the direct replacement of failed fiber sections. Investigating the feasibility of glass switched p -cycles from a technological perspective, the thesis provides an overview of recent developments in the fiber optic industry. Those developments stand in contrast with the widespread but wrong idea of very slow, bulky, expensive and size limited glass-switching fabrics; on the contrary, the new generation of whole fiber switches handles optical signals transparently and independently of data rates, formats, wavelengths, protocols and services, with faster switching times, more compact form-factors and very large demand-matrix. As well, experiments con-

ducted showed other previously unknown benefits of whole fiber switched p -cycles, which include the significant reduction of real-world CapEx costs, the removal of complexity incurred in fully transparent p -cycle designs because of wavelength continuity requirements, and linkages with PWCE which are of great interest for dynamic traffic considerations.

The remaining advancements apply to the concept of p -cycles alone. Specifically, the thesis proposes general path-protecting p -cycles (GPP) with the purpose of relaxing the failure independence constraint imposed on FIPP p -cycles, a constraint which explains the computational complexity of FIPP designs. Moreover, any GPP design solution also approximates a FIPP design because no more than three working paths remained unprotected when characterizing GPP solutions from the FIPP perspective in the experiments; but if needed, we showed how to exploit GPPs in order to derive effective FIPP solutions. Another contribution specific to p -cycles is the study of how altering grooming decisions can enhance the overall design efficiency, thereby providing a preliminary meaning to p -cycle survivable grooming.

6.3 Possible Directions for Future Work

The following research directions either pertain to limited aspects of the work done, or to real-world requirements in regard to hypotheses made within the thesis.

6.3.1 GA-ILP Improvements

Despite the ability of GA-ILP to provide high quality solutions for any large scale p -cycle network design problem, in the experiments, the GA-ILP overall process was a bit slow in general; and its convergence was quite hypothetical in such extreme case instances as huge candidate spaces combined with very large demand matrices and inherently complex problems such as FIPP p -cycles. In the future, one may envisage speeding-up the GA-ILP through a more suited implementation, which will combine AMPL/CPLEX modeling language with C++ or Java programming language. C++/Java will handle encoding, selection, crossover and mutation GA-like normal steps while AMPL/CPLEX will support the ILP-fitness based evaluation. Other GA-ILP developments can be aimed at providing a heuristic's variant fitting multi-criteria optimization contexts.

6.3.2 Multilayer Network Planning

Studies conducted within the thesis basically assumed a protection at the transport layer level. This assumption helped to develop solid insights and principles such as the two-hop flow strategy, which definitely close the debate on ordinary p -cycles' ability to recover from node failures. But why protect OXCs alone knowing that they tend to be highly robust? On the contrary, IP/MPLS routers suffer downtimes about as frequently as fiber cuts because of software patches, upgrades or even crashes. So future research studies could envisage the combined multi-layer design of node- and span-protecting p -cycles. In such a realistic design, span-protecting p -cycles will be planned in a way that provides 100% span failure protection in the optical layer while, at the same time, node-protecting p -cycles will be planned on a controlled-oversubscription basis in order to provide an MPLS-layer protection to two-hop flows transiting through a failed router.

6.3.3 Hybrid Architecture Designs

An additional research mandate consisted of unifying multi-QoP policies for p -cycle survivable networks. Doing so, we found the possibility of designing span-protecting p -cycles for a class of users with $R_{1-\text{span}}$ requirements alone, two-hop segment protecting p -cycles for another class of users surviving node and span failure events, and path-protecting p -cycles for a third class of users requiring the same services as the second class in addition to dual span failure protection and transparent reach limitations. Future work may offer a systematic study of such hybrid architectures. The main question will be “which of the span, two-hop segment, full flow and path protection schemes can be efficiently combined, from an architectural viewpoint and from a planning viewpoint?”

The *architectural viewpoint* refers to a situation where service paths that can be efficiently protected with a particular scheme were identified, then considering which of the other schemes is best suited to efficiently protect the remaining service paths. And the *planning question* will be how well the simultaneous design of schemes can be integrated, knowing that some scheme pairs are conceptually so close that design methods also seem to be similar. Hybrid design architectures may lead to some associations of schemes that work particularly well together, such as the hybrid design described for span-, two-hop segment and path-protecting p -cycles. As well, when presented with network topology and demand

information, the researcher will wonder if there will be value in a standalone “detector” tool able to identify cases in which part of the working layer can be protected with highly efficient simple structures in order to make the remainder of the network problem simpler.

6.3.4 Specific Ideas for p -Cycle Architecture Enhancements

These are remaining ideas to increase the understanding of p -cycle survivable network architectures or possibly improve the design qualities of p -cycle specific aspects developed within this thesis.

Glass Switched p -Cycles The exploratory study of p -cycle protection at a fiber-level of granularity showed many advantages from such perspectives as network transparency, real-world CapEx costs and PWCE, which may justify further investigation.

Modularity and Economy-of-scale As DWDM technology creates an extremely modular capacity-planning situation and produces potentially strong nonlinear economy-of-scale effects in capacity, future studies may investigate whether or not modularity and economy-of-scale influence the cost-optimized p -cycle design models developed in this thesis.

Joint Capacity Design Except for p -cycle survivable grooming, all other studies within the thesis assumed shortest distance weighted routing of demands under normal network conditions, performed prior to the design. This corresponds to spare capacity design (SCP) conditions, as opposed to joint working and spare capacity placement (JCP) optimizations that can be considered in further investigation.

6.4 Publications Associated with Thesis’ Research Studies

The studies associated with this thesis generated the publications below, listed per provisional patents, journal articles, conference papers, presentations and technical reports.

6.4.1 Book in Preparation

1. W.D. Grover and D.P. Ouguetou, p -Cycles: Fast, Flexible, and Efficient Network Survivability, (*a book in preparation to be published by John Wiley & Sons Inc.*)

6.4.2 Provisional Patent Applications

2. D.P. Onguetou and W.D. Grover, Using an Integer Linear Programming Model as the Fitness Function for a GA-like Heuristic, *submitted*.

6.4.3 Journal Publications

3. D.P. Onguetou and W.D. Grover, Approaches to p -Cycle Network Design with Controlled Optical Path Lengths in the Restored Network State, *OSA Journal of Optical Networking (JoN)*, vol. 7, no. 7, pp. 673–691, Jul. 2008, [OG08a].
4. D.P. Onguetou and W.D. Grover, A Simple Generalized Approach to Node Failure Recovery with Span-Protecting p -Cycles, *Journal of Networks (JNW)—Special Issue on All-Optically Routed Networks*, vol. 5, no. 11, pp. 1260–1270, Nov. 2010, [OG10b].
5. D.P. Onguetou and W.D. Grover, A Two-Hop Segment Protecting Paradigm which Unifies Node and Span Failure Protection under p -Cycles, *IEEE Communication Letters*, vol. 14, no. 11, pp. 1080–1082, Nov. 2010, [OG10c].
6. D.P. Onguetou and W.D. Grover, p -Cycle Protection at the Glass Fiber Level, *Elsevier Computer Communications Journal (COMCOM)*, vol. 34, no. 12, pp. 1399–1409, Aug. 2011, [OG11].
7. D.P. Onguetou and W.D. Grover, A GA-ILP Heuristic for Solving Large Scale p -Cycle Design Problems, *IEEE Transactions Journal*, submitted.
8. D.P. Onguetou and W.D. Grover, A Unified Approach to Multiple Quality of Protection Policies in p -Cycle Networks, *to be submitted*.

6.4.4 Peer-Reviewed Conference Papers

9. D.P. Onguetou and W.D. Grover, p -Cycle Network Design: from Fewest in Number to Smallest in Size, *Proc. of the 6th Intl. Workshop on the Design of Reliable Communication Networks (DRCN)*, La Rochelle, France, 7–10 Oct. 2007, [OG07].

10. D.P. Onguetou and W.D. Grover, A New Insight and Approach to Node Failure Protection with Ordinary p -Cycles, *Proc. of the IEEE Intl. Conf. on Communications (ICC)*, pp. 5145–5149, Beijing, China, 19–23 May 2008, [OG08b].
11. D.P. Onguetou and W.D. Grover, Solution of a 200-node Network Design Problem with GA-based Preselection of Candidate Structures, *Proc. of the IEEE Intl. Conf. on Communications (ICC)*, Dresden, Germany, 14–18 Jun. 2009, [OG09b].
12. W.D. Grover and D.P. Onguetou, A New Approach to Node Failure Protection with Span-Protecting p -Cycles, *Proc. of 11th Intl. Conf. on Transparent Optical Networks—5th Workshop on Reliability Issues in Next Generation Optical Networks (ICTON /RONEXT)*, Island of Sao Miguel (Azores), Portugal, 28 Jun.–2 Jul. 2009, [GO09].
13. D.P. Onguetou and W.D. Grover, Whole Fiber Switched p -Cycles, *Proc. of the Seventh Intl. Workshop on the Design of Reliable Communication Networks (DRCN)*, pp. 55–61, Wash. DC area, USA, 25–28 Oct. 2009, [OG09c].
14. D.P. Onguetou, Dimitri Baloukov and W.D. Grover, Near-Optimal FIPP p -Cycle Network Designs using General Path-Protecting p -Cycles and Combined GA-ILP Methods, *Proc. of the 7th Intl. Workshop on the Design of Reliable Communication Networks (DRCN)*, pp. 243–250, Wash. DC area, USA, 25–28 Oct. 2009, [OBG09].
15. D.P. Onguetou and W.D. Grover, Altering Grooming Decisions to Enhance p -Cycle Design Efficiency, *Proc. of the IEEE Global Telecommunications Conference (GLOBECOM)*, Honolulu, Hawaii, USA, 30 Nov.–4 Dec. 2009, [OG09a].
16. A. Grue, W.D. Grover, M. Clouqueur, D.A. Schupke, J. Doucette, B. Forst, D.P. Onguetou and D. Baloukov, Comparative Study of Fully Pre-Cross-Connected Protection Architectures for Transparent Optical Networks, *Proc. of the 6th Intl Workshop on the Design of Reliable Communication Networks (DRCN)*, La Rochelle, France, 7–10 Oct. 2007, [GGC⁺07].
17. A. Grue, W.D. Grover, M. Clouqueur, D.A. Schupke, D. Baloukov, D.P. Onguetou and B. Forst, Capex Costs of Lightly Loaded Restorable Networks under a Consistent

WDM Layer Cost Model, *Proc. of the IEEE Intl. Conf. on Communications (ICC)*, Dresden, Germany, 14–18 Jun. 2009, [GGC⁺09].

6.4.5 Additional Presentations and Technical Reports

18. W.D. Grover and D.P. Onguetou, Towards Solution of Large-Scale p -Cycle Network Design Problems with a Combined GA-ILP Heuristic, *5th Workshop on Optimization of Optical Networks (OON)*, Montréal, Québec, Canada, 7–8 May 2008, [GO08].
19. A. Grue, D.P. Onguetou and W.D. Grover, Study of p -Cycle Designs for South Africa Telecommunication Network, *Technical Memo*, Oct. 2006.
20. A. Grue, B. Forst, D.P. Onguetou, D. Baloukov, J. Doucette, A. Kodian and W.D. Grover, Project HAVANA: Comparative Study of Fully Pre-cross-connected Protection Architectures for High Availability Transparent Optical Networking, *TRLabs Technical Report ST-06-01*, Edmonton, Canada, 9 Nov. 2006.
21. A. Grue, B. Forst, D.P. Onguetou, D. Baloukov, J. Doucette, A. Kodian and W.D. Grover, First-year Slide Decks, HAVANA, 2006.
22. A. Grue, B. Forst, D.P. Onguetou, D. Baloukov and W.D. Grover, Second-year Slide Decks Produced for Nokia Siemens Networks, HAVANA, 2007.
23. Aden Grue, W.D. Grover, D. Baloukov and D.P. Onguetou, High Availability Network Architectures (HAVANA): Application of the NOBEL Cost Model, *Invited talk given to Nokia Siemens Networks*, Munich, Germany, 19 Sep. 2008.
24. A. Grue, W.D. Grover, D. Baloukov and D.P. Onguetou, High Availability Network Architectures (HAVANA): Overview and Wrap-up, Edmonton, Canada, 10 Dec. 2008.
25. A. Grue, B. Forst, D.P. Onguetou, D. Baloukov and W.D. Grover, Third-year Slide Decks, HAVANA, 2008.

Bibliography

- [Bal09] Dimitri Baloukov. Advances in Failure Independent Path-Protecting p -Cycle Network Design. Master's thesis, University of Alberta, Department of Electrical and Computer Engineering, Edmonton, AB, Canada, Fall 2009.
- [BGK07] Dimitri Baloukov, Wayne D. Grover, and Adil Kodian. Towards Jointly Optimized Design of Failure Independent Path Protecting p -Cycle Networks. *OSA Journal of Optical Networking (JoN)*, 6(12):62–79, Dec. 2007.
- [BJ10] Loudon Blair and Chris Janson. The Next Step in the Optical Network Evolution: OTN. Ciena Virtual Seminar, Feb. 9th 2010.
- [BRS04] G. Bernstein, B. Rajagopalan, and D. Saha. *Optical Network Control: Architecture, Protocols and Standards*. Addison Wesley, 2004.
- [BSGN03] F.J. Blouin, Anthony Sack, Wayne D. Grover, and H. Nasrallah. Benefits of p -Cycles in a Mixed Protection and Restoration Approach. In *Proceedings of the 4th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 203–211, Banff, Alberta, Canada, Oct. 19-22 2003.
- [CCF04] T.Y. Chow, F. Chudak, and A.M. Ffrench. Fast Optical Layer Mesh Protection Using Pre-Cross-Connected Trails. *IEEE/ACM Transactions on Networking*, 12(3):539–547, Jun. 2004.
- [CG05] Matthieu Clouqueur and Wayne D. Grover. Availability Analysis and Enhanced Availability Design in p -Cycle-Based Networks. *Photonic Network Communications, Springer Science*, 10(1):55–71, 2005.

- [Cra93] D. Crawford. Fiber Optic Cable Dig-ups “causes and cures”. In *Network Reliability: A Report to the Nation—Compendium of Technical Papers*, Chicago, Jun. 1993. National Engineering Consortium.
- [CWN⁺02] D. Chen, S. Wheeler, D. Nguyen, B. Davis, M. Glavanovic, J. Khaydarov, I. Koruga, S. Hegarty, F. Cokic, and Fei Zhu. 40 Channels 4000 km DWDM ULH Transmission Field Trial without Raman Amplification and Regeneration. In *Proceedings of Optical Fiber Communication Conference (OFC)*, pages FC10–1—FC10–3, 2002.
- [CZKC] E. Carr, P. Zhang, D. Keebaugh, and K. Chau. Cost-Effective Method of Manufacturing a 3D MEMS Optical Switch. In *Proceedings of SPIE*, volume 7204.
- [DC09] Robert Doverspike and Bruce Cortez. Restoration in Carrier Networks. In *Proceedings of the 7th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 45–54, Washington DC area, USA, 25-28 Oct. 2009.
- [DG03] John E. Doucette and Wayne D. Grover. Node-Inclusive Span Survivability in an Optical Mesh Transport Network. In *Proceedings of the 19th annual National Fiber Optics Engineers Conference (NFOEC)*, pages 634–643, Orlando, Sep. 2003.
- [DGG05] John E. Doucette, Peter A. Giese, and Wayne D. Grover. Combined Node and Span Protection Strategies with Node-Encircling p -Cycles. In *Proceedings of the 5th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 213–221, Ishia (Naples), Italy, 16-19 Oct. 2005.
- [DGG07] John E. Doucette, Wayne D. Grover, and Peter A. Giese. Physical-Layer p -Cycle Adapted for Router-Level Node Protection: a Multi-Layer Design and Operation Strategy. *IEEE Journal on Selected Areas in Communications (JSAC)*, 25(5):963–973, Jun. 2007.
- [DHGY03] John E. Doucette, Donna He, Wayne D. Grover, and Oliver Yang. Algorithmic Approaches for Efficient Enumeration of Candidate p -Cycles and Capacitated

- p*-Cycle Network Design. In *Proceedings of the 4th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 212–220, Banff, Alberta, Canada, Oct. 2003.
- [Dou01] John E. Doucette. SCP *p*-Cycle AMPL Data File Created for Use with PCycle-SCP.mod AMPL Model, Dec. 2001.
- [Dou06] John E. Doucette. Slide Decks for EE-780 Class on Network Optimization and Reliability, Winter 2006.
- [DSCB04] W. C. Dickson, B. P. Staker, G. Campbell, and W. C. Banyai. 64x64 3D-MEMS Switch Control System with Robustness to MEMS Resonant Frequency Variation and Pointing Drift. Technical Digest ThQ5, OFC, 2004.
- [EGRG05] Manohar Naidu Ellanti, Steven Scott Gorshe, Lakshmi G. Raman, and Wayne D. Grover. *Next Generation Transport Networks: Data, Management, and Control Planes*. Springer, 2005.
- [FG07] Brian Forst and Wayne D. Grover. Factors Affecting the Efficiency of Demand-wise Shared Protection. In *Proceedings of the 6th International Workshop on the Design of Reliable Communication Networks (DRCN)*, La Rochelle, France, October 2007.
- [For09] Brian David Forst. Analysis and Understanding of Demand-wise Shared Protection. Master’s thesis, University of Alberta, Department of Electrical and Computer Engineering, Edmonton, AB, Canada, Spring 2009.
- [FSO⁺] A. Fernandez, B. P. Staker, W. E. Owens, L. P. Murray, J. P. Spallas, and W. C. Banyai. Modular MEMS Design and Fabrication for an 80x80 Transparent Optical Cross-Connect Switch. In *Proceedings of the SPIE*, volume 5604, pages 208–217.
- [Gar] Gartner Group. <http://www.gartner.com/>, (lastchecked: Jul. 8th, 2010).
- [GCM⁺03] A. Groebbens, D. Colle, S. D. Maeschaleck, I. Lievens, M. Pickavet, and P. Demeester. Efficient Protection in MPAS Networks using Backup Trees: part

- one-concepts and heuristics. *Journal of Photonic Network Communications*, 6:191–206, 2003.
- [GD02] Wayne D. Grover and John E. Doucette. Advances in Optical Network Design with p -Cycles: Joint Optimization and Preselection of Candidate p -Cycles. In *Proceedings of IEEE/LEOS Summer Topicals 2002*, pages 49–50, Mont-Tremblant, Québec, Canada, Jul. 2002.
- [GFO⁺06] A. Grue, B. Forst, D.P. Onguetou, D. Baloukov, J. Doucette, A. Kodian, and W.D. Grover. Project HAVANA: Comparative Study of Fully Pre-cross-connected Protection Architectures for High Availability Transparent Optical Networking. Technical Report ST-06-01, TRILabs, Edmonton AB, Canada, Nov. 9 2006.
- [GG06] Aden Grue and Wayne D. Grover. Characterization of Pre-Cross-Connected Trails for Optical Mesh Network Protection. *OSA Journal of Optical Networking (JoN)*, 5(6):493–508, June 2006.
- [GG07a] Wayne D. Grover and Aden Grue. Self-Fault Isolation in Transparent p -Cycle networks: p -Cycles as their Own m -Cycles. *IEEE Communication Letters*, 11(12):1004–1006, December 2007.
- [GG07b] Aden Grue and Wayne D. Grover. Comparison of p -Cycles and p -Trees in a Unified Framework. *Springer Photonic Network Communications*, 14(2):123–134, Oct. 2007.
- [GG07c] Aden Grue and Wayne D. Grover. Improved Method for Survivable Network Design based on Pre-Cross-Connected Trails. *OSA Journal of Optical Networking (JoN)*, 6(2):200–216, February 2007.
- [GG09] Aden Grue and Wayne D. Grover. UPSR-like p -Cycles: A New Approach to Dual Failure Protection. In *Proceedings of the Workshop on Reliable Network Design and Modeling (RNDM), part of the International Conference on Ultra Modern Telecommunications*, St. Petersburg, Russia, Oct. 12-14 2009.

- [GGC⁺07] Aden Grue, Wayne D. Grover, Matthieu Clouqueur, Dominic A. Schupke, John E. Doucette, Brian Forst, Diane Prisca Onguetou, and Dimitri Baloukov. Comparative Study of Fully Pre-Cross-Connected Protection Architectures for Transparent Optical Networks. In *Proceedings of the 6th International Workshop on the Design of Reliable Communication Networks (DRCN)*, La Rochelle, France, 7-10 Oct. 2007.
- [GGC⁺09] Aden Grue, Wayne D. Grover, Matthieu Clouqueur, Dominic A. Schupke, Dimitri Baloukov, Diane Prisca Onguetou, and Brian Forst. Capex Costs of Lightly Loaded Restorable Networks Under a Consistent WDM Layer Cost Model. In *Proceedings of the IEEE International Conference on Communications (ICC)*, Dresden, Germany, 14-18 Jun. 2009.
- [GK05] Wayne D. Grover and Adil Kodian. Failure-Independent Path Protection with p -Cycles: Efficient, Fast and Simple Protection for Transparent Optical Networks. In *Proceedings of 7th International Conference on Transparent Optical Networks (ICTON'05)*, Barcelona, Spain, 3-7 Jul. 2005.
- [GKO⁺05] C. G. Gruber, A. M. C. A. Koster, S. Orłowski, R. Wessälly, and A. Zymolka. A Computational Study for Demand-wise Shared Protection. In *Proceedings of the 5th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 421–428, Naples, Italy, October 2005.
- [Glia] Glimmerglass Networks. <http://www.glimmerglass.com/index.aspx>. (Lastchecked may 17th, 2011).
- [Glib] Glimmerglass Networks. Intelligent Optical Switches: Product Family Overview.
- [GLW⁺06] M. Gunkel, R. Leppla, M. Wade, A. Lord, Dominic A. Schupke, G. Lehmann, C. Fürst, S. Bodamer, B. Bollenz, H. Haunstein, H. Nakajima, and J. Martensson. A Cost Model for the WDM Layer. In *International Conference on Photonics in Switching (PS)*, Herakleion (Crete), Greece, Oct. 2006.
- [GO08] Wayne D. Grover and Diane Prisca Onguetou. Towards Solution of Large-Scale p -Cycle Network Design Problems with a Combined GA-ILP Heuristic.

- Key Note Speech at the Fifth Workshop on Optimization of Optical Networks (OON), May 2008.
- [GO09] Wayne D. Grover and Diane Prisca Onguetou. A New Approach to Node-Failure Protection with Span-Protecting p -Cycles. In *Proceedings of the 11th International Conference on Transparent Optical Networks–5th Workshop on Reliability Issues in Next Generation Optical Networks (ICTON/RONEXT)*, Island of Sao Miguel (Azores), Portugal, 28Jun.–2Jul. 2009.
- [Gro02a] Wayne D. Grover. Slide Decks for ECE-681 Class on Survivable Transport Networks, Fall 2002.
- [Gro02b] Wayne D. Grover. Understanding p -Cycles, Enhanced Rings, and Oriented Cycle Covers. In *Proceedings of the 1st Intl. Conference on Optical Communications and Networks (ICOON)*, pages 305–308, Singapore, Nov. 11-14 2002.
- [Gro03a] Wayne D. Grover. p -Cycles, Ring-Mesh Hybrids and "Ring-Mining:" Options for New and Evolving Optical Networks. In *Proceedings of Optical Fiber Communication Conference (OFC)*, pages 201–203, Atlanta, Georgia (USA), Mar. 24-27 2003.
- [Gro03b] Wayne D. Grover. *Mesh-Based Survivable Networks: Options and Strategies for Optical, MPLS, SONET, and ATM Networking*. PTR Prentice Hall, 2003.
- [Gro05] Wayne D. Grover. Path-Protecting p -Cycles and the Protected Working Capacity Envelope Concept: how they address the needs of dynamic transparent optical networks. In *Proceedings of the SPIE Photonics*, Toronto, ON, Canada, Sep. 12-14 2005.
- [Gru09] Aden Justus Grue. *Pre-Cross-Connected Protection Architectures for Transparent Optical Transport Networks*. PhD thesis, University of Alberta, Department of Electrical and Computer Engineering, Edmonton, AB, Canada, Fall 2009.
- [GS98a] Wayne D. Grover and Demetrios Stamatelakis. Cycle-oriented Distributed Preconfiguration: Ring-like Speed with Mesh-like Capacity for Self-planning

- Network Restoration. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 537–543, Atlanta, Georgia (USA), 8-11 Jun. 1998.
- [GS98b] Wayne D. Grover and Demetrios Stamatelakis. Self-organizing Closed-Path Configuration of Spare Capacity in Broadband Mesh Transport Networks. In *Proceedings of the Canadian Conference on Broadband Research (CCBR)*, pages 145–156, Ottawa, ON, Canada, Jun. 12-24 1998.
- [GS00] Wayne D. Grover and Demetrios Stamatelakis. Bridging the Ring-Mesh Dichotomy with p -Cycles”. In *Proceedings of the IEEE/VDE Design of Reliable Communication Networks (DRCN)*, pages 181–186, Munich, Germany, Apr. 2000.
- [HGMS08] R. Huelsermann, M. Gunkel, C. Meusburger, and Dominic A. Schupke. Cost Modelling and Evaluation of Capital Expenditures in Optical Multilayer Networks. *OSA Journal of Optical Networking (JoN)*, 7(9):814–833, Sep. 2008.
- [HJK⁺06] R. Hülsermann, M. Jäger, A. M. C. A. Koster, S. Orłowski, R. Wessälly, and A. Zymolka. Availability and Cost-based Evaluation of Demand-wise Shared Protection. In *Proceedings of the 7th ITG-Workshop on Photonic Networks*, pages 161–168, Leipzig, Germany, 2006.
- [Joh75] D.B. Johnson. Finding All Elementary Circuits of a Directed Graph. *SIAM Journal on Computing*, 4:77–84, 1975.
- [JP02] A. J.Vernon and J. D. Portier. Protection of Optical Channels in All-Optical Networks. In *18th Annual National Fiber Optic Engineers Conference (NFOEC)*, pages 1695–1706, Dallas, Texas, USA, Sep. 2002.
- [JRBG07] Brigitte Jaumard, Caroline Rocha, Dimitri Baloukov, and Wayne D. Grover. A Column Generation Approach for Design of Networks using Path-Protecting p -Cycles. In *Proceedings of the 6th International Workshop on the Design of Reliable Communication Networks (DRCN)*, La Rochelle, France, 7-10 Oct. 2007.

- [JSR09] Brigitte Jaumard, Samir Sebbah, and Caroline Rocha. Large Scale Optimization in Survivable WDM Networks. In *Tutorials of the 7th International Workshop on the Design of Reliable Communication Networks (DRCN)*, Washington DC area, USA, Oct. 2009.
- [KG05a] Adil Kodian and Wayne D. Grover. Failure Independent Path-Protecting p -Cycles: Efficient and Fully Pre-Connected Optical-Path Protection. *IEEE/OSA Journal of Lightwave Technology (JLT)*, 23(10), Oct. 2005.
- [KG05b] Adil Kodian and Wayne D. Grover. Multiple-Quality of Protection Classes Including Dual-Failure Survivable Services in p -Cycle networks. In *Proceedings of the Second IEEE International Conference on Broadband Networks (BroadNets)—Optical Networking Symposium*, Boston, Oct. 2005.
- [KGD05] Adil Kodian, Wayne D. Grover, and John E. Doucette. A Disjoint Route Sets Approach to Design of Failure-Independent Path-Protecting p -Cycle Networks. In *Proceedings of the 5th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 231–238, Ishia (Naples), Italy, 16-19 Oct. 2005.
- [KGSM03] Adil Kodian, Wayne D. Grover, J. Slevinsky, and D. Moore. Ring-Mining to p -Cycles as a Target Architecture: Riding Demand Growth into Network Efficiency. In *Proceedings of the 19th annual National Fiber Optics Engineers Conference (NFOEC)*, pages 1543–1552, Orlando (USA), Sep. 2003.
- [KSG04] Adil Kodian, Anthony Sack, and Wayne D. Grover. p -Cycle Network Design with Hop Limits and Circumference Limits. In *Proceedings of the 1st International Conference on Broadband Networks (BROADNETS)*, pages 244–253, San Jose, California (USA), Oct. 25-29 2004.
- [KSG05] Adil Kodian, Anthony Sack, and Wayne D. Grover. The Threshold Hop-Limit Effect in p -Cycles: Comparing Hop- and Circumference Limited Design. *Optical Switching Networking (OSN)*, Elsevier, pages 72–85, Jul. 2005.

- [KZJH05] A. M. C. A. Koster, A. Zymolka, M. Jager, and R. Hulsermann. Demand-wise Shared Protection for Meshed Optical Networks. *Journal of Network Systems Management*, 13(1):35–55, Mar. 2005.
- [LG04] Dion Leung and Wayne D. Grover. Capacity Design of p -Cycle Networks in the Face of Demand Forecast Uncertainty. In *Proceedings of 9th OptoElectronics and Communications Conference and 3rd International Conference on Optical Internet (OECC/COIN)*, Yokohama, Japan, Jul. 12-16 2004.
- [MD76] P. Mateti and N. Deo. On Algorithms for Enumerating All Circuits of a Graph. *SIAM Journal on Computing*, 5(1):90–99, Mar. 1976.
- [Mor01] George David Morley. *Analysis and Design of Ring-based Transport Networks*. Phd thesis, Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada, Spring 2001.
- [Nob] IST Project NOBEL, <http://www.ist-nobel.org>, (lastchecked: Jan. 14th, 2010).
- [OBG09] Diane Prisca Onguetou, Dimitri Baloukov, and Wayne D. Grover. Near-Optimal FIPP p -Cycle Network Designs using General Path-Protecting p -Cycles and Combined GA-ILP Methods. In *Proceedings of the 7th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 243–250, Washington DC area, USA, Oct. 2009.
- [OG07] Diane Prisca Onguetou and Wayne D. Grover. p -Cycle Network Design: from Fewest in Number to Smallest in Size. In *Proceedings of the 6th International Workshop on the Design of Reliable Communication Networks (DRCN)*, La Rochelle, France, 7-10 Oct. 2007.
- [OG08a] Diane Prisca Onguetou and Wayne D. Grover. Approaches to p -Cycle Network Design with Controlled Optical Path Lengths in the Restored Network State. *OSA Journal of Optical Networking (JoN)*, 7(7):673–691, July 2008.
- [OG08b] Diane Prisca Onguetou and Wayne D. Grover. A New Insight and Approach to Node Failure Protection with Ordinary p -Cycles. In *Proceedings of the*

- IEEE International Conference on Communications (ICC)*, pages 5145–5149, Beijing, China, 19-23 May 2008.
- [OG09a] Diane Prisca Onguetou and Wayne D. Grover. Altering Grooming Decisions to Enhance p -Cycle Design Efficiency. In *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, Honolulu, Hawaii, USA, Nov. 30-Dec. 4 2009.
- [OG09b] Diane Prisca Onguetou and Wayne D. Grover. Solution of a 200-node Network Design Problem with GA-based Preselection of Candidate Structures. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 1–5, Dresden, Germany, Jun. 2009.
- [OG09c] Diane Prisca Onguetou and Wayne D. Grover. Whole Fiber Switched p -Cycles. In *Proceedings of the 7th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 55–61, Washington DC area, USA, 25-28 Oct. 2009.
- [OG10a] Diane Prisca Onguetou and Wayne D. Grover. A GA-ILP Heuristic for Solving Large Scale p -Cycle Design Problems. *submitted to IEEE Transactions...*, 2010.
- [OG10b] Diane Prisca Onguetou and Wayne D. Grover. A Simple Generalized approach to Node Failure Recovery with Span-Protecting p -Cycles. *In Press with Journal of Networks Special Issue on All-Optically Routed Networks*, 5(11):1260–1270, Jun. 2010.
- [OG10c] Diane Prisca Onguetou and Wayne D. Grover. A Two-Hop Segment Protecting Paradigm which Unifies Node and Span Failure Protection under p -Cycles. *submitted to IEEE Communication Letters*, 14(11):1080–1082, Nov. 2010.
- [OG11] Diane Prisca Onguetou and Wayne D. Grover. p -Cycle Protection at the Glass Fiber Level. *Elsevier Computer Communications (COMCOM) Journal*, 34(12):1399–1409, Aug. 2011.
- [SCG04] Dominic A. Schupke, Matthieu Clouqueur, and Wayne D. Grover. Strategies for Enhanced Dual Failure Restorability with Static or Reconfigurable p -Cycle

- Networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, Paris, France, Jun. 2004.
- [Sch04] Dominic A. Schupke. An ILP for Optimal p -Cycle Selection without Cycle Enumeration. In *Proceedings of the 8th Conference on Optical Network Design and Modelling (ONDM)*, Ghent, Belgium, Feb. 2004.
- [Sch05] Dominic A. Schupke. Automatic Protection Switching for p -Cycles in WDM Networks. *Optical Switching Networking (OSN)*, Elsevier, 2(1):35–48, May 2005.
- [SG99] Demetrios Stamatelakis and Wayne D. Grover. Rapid Restoration of Internet Protocol Networks using Pre-configured Protection Cycles. In *Proceedings of the 3rd Canadian Conference on Broadband Research (CCBR)*, pages 33–44, Ottawa, ON, Canada, Nov. 7-9 1999.
- [SG00a] Demetrios Stamatelakis and Wayne D. Grover. IP Layer Restoration and Network Planning based on Virtual Protection Cycles. *IEEE Journal on Selected Areas in Communications (JSAC)–Special Issue on Protocols and Architectures for Next Generation Networks*, 18(10):1912–1913, Oct. 2000.
- [SG00b] Demetrios Stamatelakis and Wayne D. Grover. Theoretical Underpinnings for the Efficiency of Restorable Networks using Pre-configured Cycles (“ p -cycles”). *IEEE Transactions on Communications*, 48(8):1262–1265, Aug. 2000.
- [SG03a] Gangxiang Shen and Wayne D. Grover. Exploiting Forcer Structure to Serve Uncertain Demands and Minimize Redundancy of p -Cycle Networks. In *Proceedings of Optical Networking and Communications (OptiComm)*, pages 59–70, Dallas, Texas, USA, Oct. 13-17 2003.
- [SG03b] Gangxiang Shen and Wayne D. Grover. Extending the p -Cycle Concept to Path-Segment Protection for Span and Node Failure Recovery. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 1314–1319, Anchorage (Alaska), USA, 11-15 May 2003.

- [SG03c] Gangxiang Shen and Wayne D. Grover. Extending the p -Cycle Concept to Path-Segment Protection for Span and Node Failure Recovery. *IEEE Journal on Selected Areas in Communications (JSAC)*, 21(8):1306–1319, Oct. 2003.
- [SG04a] Anthony Sack and Wayne D. Grover. Hamiltonian p -Cycles for Fiber-level Protection in Homogeneous and Semi-Homogeneous Optical Networks. *IEEE Network, Special Issue on Protection, Restoration and Disaster Recovery*, 18(2):49–56, Mar-Apr. 2004.
- [SG04b] Gangxiang Shen and Wayne D. Grover. Performance of Protected Working Capacity Envelopes based on p -Cycles: Fast, Simple, and Scalable Dynamic Service Provisioning of Survivable Services. In *Proceedings SPIE Asia-Pacific Optical Communications Conference on Network Architectures, Management, and Applications II (APOC)*, volume 5626, pages 519–533, Beijing, China, Nov. 7-11 2004.
- [SG05] Gangxiang Shen and Wayne D. Grover. Design and Performance of Protected Working Capacity Envelopes based on p -Cycles for Dynamic Provisioning of Survivable Services. *OSA Journal of Optical Networking (JoN)*, 4(7):361–390, Jul. 2005.
- [SGA02] Dominic A. Schupke, C. G. Gruber, and A. Autenrieth. Optimal Configuration of p -Cycles in WDM Networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 2761–2765, New York City, NY (USA), 2002.
- [SHY04] S. Shah-Heydi and O. Yang. Hierarchical Protection Tree Scheme for Failure Recovery in Mesh Networks. *Journal of Photonic Network Communications*, 7:145–159, 2004.
- [SND06] SNDLib. <http://sndlib.zib.de/>, 2006.
- [SSG03a] Dominic A. Schupke, Matthias C. Sheffel, and Wayne D. Grover. Configuration of p -Cycles in WDM Networks with Partial Wavelength Conversion. *Photonic Network Communications, Kluwer Academic*, 6(3):239–252, Nov. 2003.

- [SSG03b] Dominic A. Schupke, Matthias C. Sheffel, and Wayne D. Grover. An efficient strategy for Wavelength Conversion in WDM p -Cycle Networks. In *Proceedings of the 4th International Workshop on the Design of Reliable Communication Networks (DRCN)*, pages 221–227, Banff, Alberta, Canada, Oct. 2003.
- [ST04] Thomas Stiden and Tommy Thomadsen. Joint Optimization of Working and p -Cycle Protection Capacity. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DK-2800 Kongens Lyngby, Denmark, May 2004.
- [ST05] Thomas Stidsen and Tommy Thomadsen. Joint Routing and Protection Using p -Cycles. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DK-2800 Kongens Lyngby, Denmark, May 2005.
- [TN94] M. To and P. Neusy. Unavailability Analysis of Long-Haul Networks. *IEEE Journal on Selected Areas in Communications (JSAC)*, 12(1):100–109, Jan. 1994.
- [TS03] Tommy Thomadsen and Thomas Stiden. The Quadratic Selective Travelling Salesman Problem. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DK-2800 Kongens Lyngby, Denmark, 2003.
- [WOZ⁺05] R. Wessály, S. Orlowski, A. Zymolka, A. M. C. A. Koster, and C. G. Gruber. Demand-wise Shared Protection revisited: a new model for survivable network design. In *Proceedings of the International Network Optimization Conference (INOC)*, pages 100–105, Lisbon, Portugal, March 2005.
- [WYH07] Bin Wu, Kwan L. Yeung, and Pin-Han Ho. ILP Formulations for p -Cycle Design without Candidate Cycle Enumeration. In *Proceedings of the IEEE International Conference on Communications (ICC)*, Glasgow, Scotland, Jun. 2007.
- [ZY02] H. Zhang and O. Yang. Finding Protection Cycles in DWDM Networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, volume 5, pages 2756–2760, New York City, NY (USA), 28 Apr.–2 May 2002.

Appendices

Appendix A

Portrayal of a Conventional p -Cycle Design

The following pictures describe a 100% span restorable design using p -cycles. Working capacities to be protected in the event of failure are indicated at top spans and spare capacities building the p -cycles involved in the solution are shown on edges in another graph. All p -cycles are given in one graph showing how spare capacity requirements are derived.

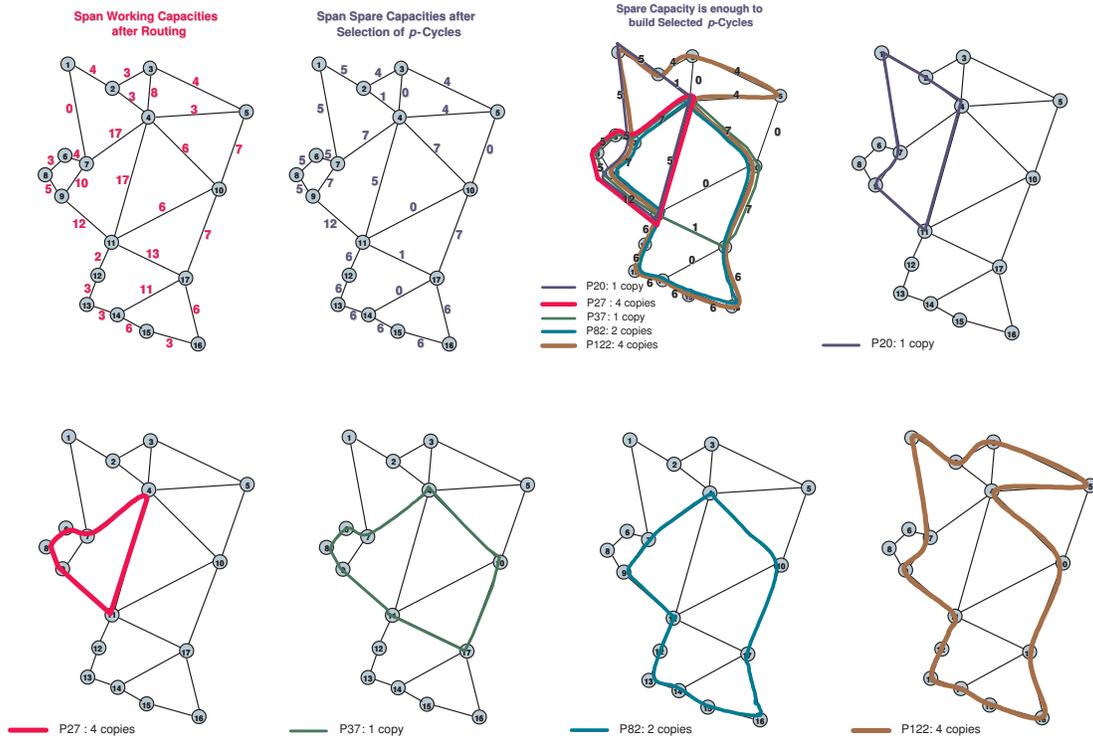
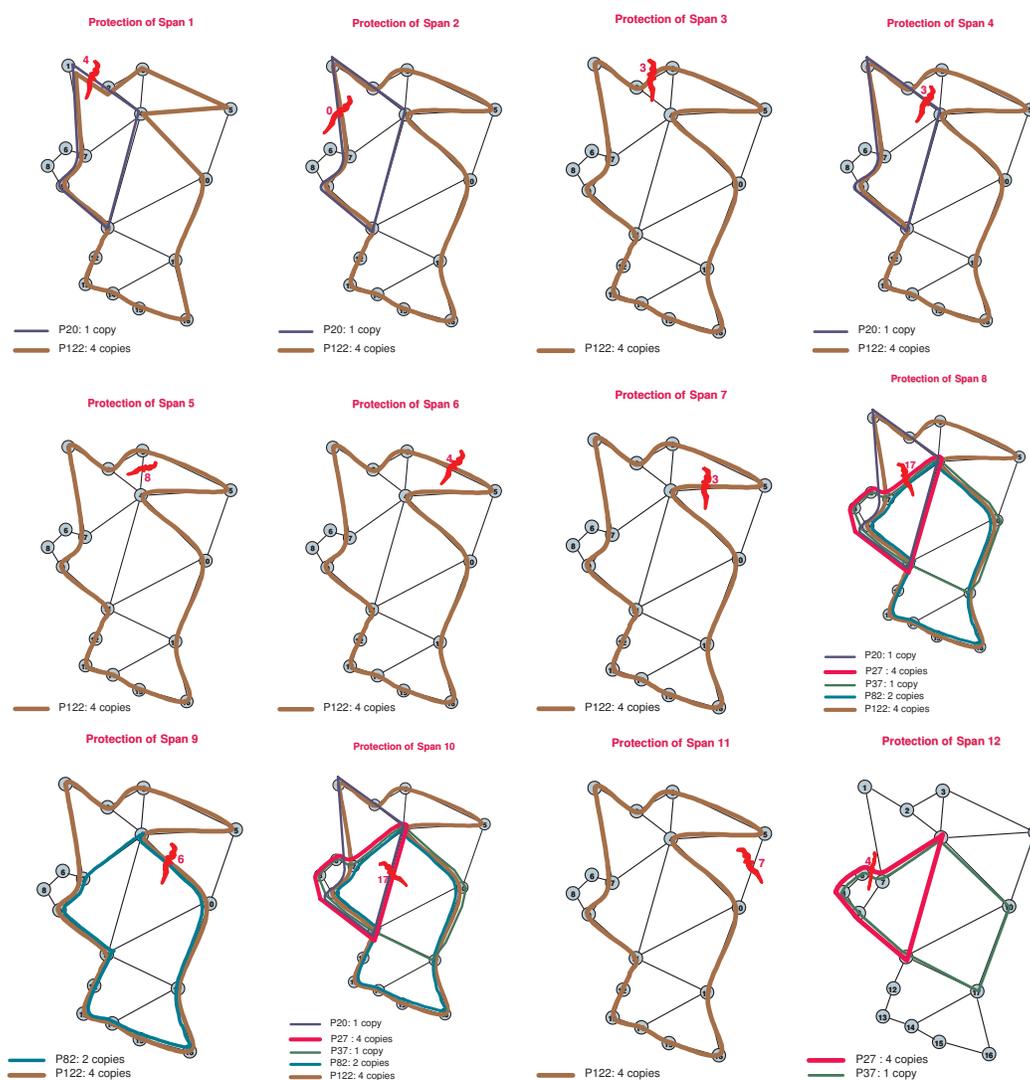


Figure A.1: Network Solution and p -Cycle Selection

A. Portrayal of a Conventional p -Cycle Design

For more clarity, each of the cycle structure involved in the design is now extracted by itself in other graphs. Then, the effectiveness of the overall design is illustrated for every span across the network, showing that the number of working channels riding on the span in question is lower than the number of protection segments through p -cycles available to protect that span. In interpreting the proof of effectiveness, recall that a unit-sized a single unit-sized copy of p -cycle handles one working channel for every on-cycle span and two protection channels for each of the straddling spans.



A. Portrayal of a Conventional p -Cycle Design

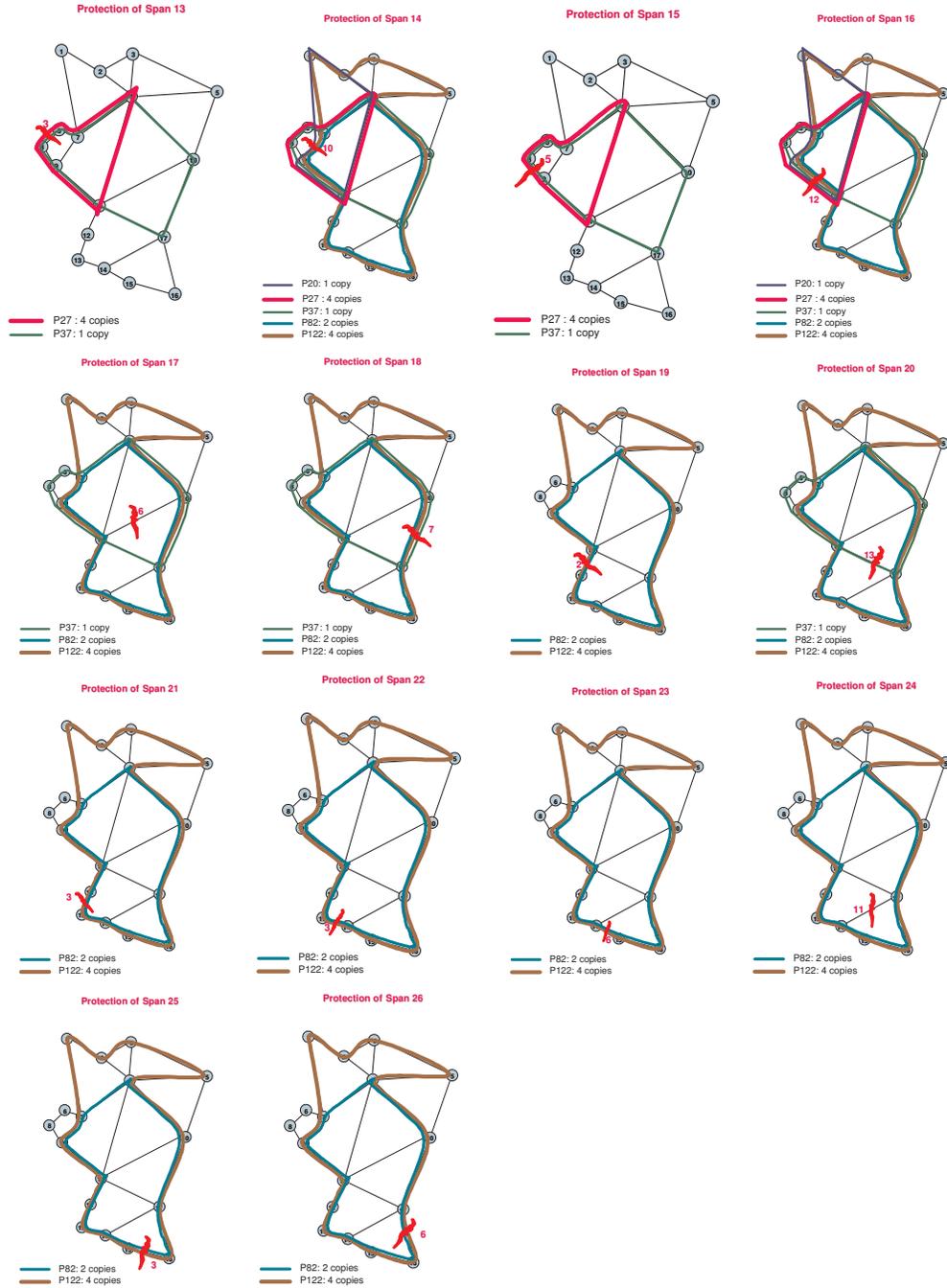


Figure A.2: Proof of Effectiveness

Appendix B

ILP Design Models for Wavelength Switched p -Cycles

[SSG03a] distinguishes between three types of p -cycle configuration in the WDM layer: i.e., opaque, hybrid and fully transparent p -cycle designs.

B.1 Symbology

Sets

- S is the set of spans, indexed by i for failing spans and j for surviving or spans in general.
- P is the set of eligible cycles, determined by a pre-processing method and indexed by p .
- C is the set of wavelengths available into a fiber. All fiber optics across a given network are assumed the same waveband of $|C|$ wavelengths, $|C|$ being the cardinality of set C . Previous sets, S and P , are considered as well.
- D represents the set of demands (indexed by r).

Input Parameters

- C_j is the cost of each capacity unit on span j . But in upcoming CapEx cost considerations, C_j will also refer to the length of span j .
- w_i is the number of working channels (capacity units) to be protected on span i , and which arise from whatever routing of the demand matrix is employed.
- d^r is the number of units of capacity for demand relation r .

B. ILP Design Models for Wavelength Switched p -Cycles

- $\delta_j^r \in \{0, 1\}$ indicates spans that each given demand relation r crosses en route; $\delta_j^r = 1$ if r crosses j and $\delta_j^r = 0$ otherwise. Also, all units for a given demand relation r are assumed to follow the same working route under normal network states.
- $x_i^p \in \{0, 1, 2\}$ encodes the number of protection route-segments that one unit-sized copy of p -cycle p provides to span i . $x_i^p = 2$ if span i straddles p -cycle p , $x_i^p = 1$ if span i is on p -cycle p and $x_i^p = 0$ otherwise.

Decision Variables

- s_j is the total number of spare channels needed on span j in the design.
- η^p is the number of unit-channel copies of candidate p -cycle p used in the design.
- F_j encodes the number of bidirectional fibers which comprise span j . In some variant, F_j is now broken into F_{w_j} and F_{s_j} in order to separate working and protection fibers in the design.
- $\lambda_l^r \in \{0, 1\}$ records wavelengths assigned to working routes for demand relation r ; $\lambda_l^r = 1$ if r uses wavelength l under normal network states and $\lambda_l^r = 0$ otherwise.
- $\zeta_l^p \in \{0, 1\}$ similarly records wavelengths assigned to cycle structure p in the design; $\zeta_l^p = 1$ if p -cycle p uses wavelength l and $\zeta_l^p = 0$ otherwise.

Decision variables λ_l^r and ζ_l^p are assumed binary to simplify the ILP formulation. This condition imposes the total number of units for a given demand relation or cycle structure to be less than or equal to the number of wavelengths available into each fiber optic.

B.2 ILP for Opaque Wavelength Switched p -Cycles

Equations (B.1)–(B.4) define the ILP design model for opaque wavelength switched p -cycles. The bi-criterion objective (B.1) minimizes total spare capacity requirements and bias the design towards choosing p -cycles that will minimize the total number of fiber optics. Constraints (B.2) and (B.4) ensure 100% restorability against single span failures

B. ILP Design Models for Wavelength Switched p -Cycles

through p -cycles that are built in the spare capacity. And equation (B.3) computes the number of fibers required in the design.

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j + \alpha \cdot \sum_{j \in S} F_j. \quad (\text{B.1})$$

$$w_i \leq \sum_{p \in P} x_i^p \cdot \eta^p, \quad \forall i \in S. \quad (\text{B.2})$$

$$F_j \geq \frac{w_j + s_j}{|C|}, \quad \forall j \in S. \quad (\text{B.3})$$

$$s_j = \sum_{p \in P: x_j^p=1} \eta^p, \quad \forall j \in S. \quad (\text{B.4})$$

B.3 ILP Formulation for Hybrid p -Cycle Designs

The ILP design model for hybrid p -cycle configurations is given by equations (B.2)–(B.8). This ILP re-uses all equations available for the opaque case, *plus* four additional sets of constraints pertaining to wavelength continuity requirements along working paths and p -cycles. More specifically, equations (B.5) and (B.6) assign a single wavelength to every working path and to every unit-sized copy of p -cycle in the design. Equation (B.7) ensures that each specific wavelength is assigned no more than once into any fiber optic and equation (B.8) makes sure that overall, wavelength allocation not exceed the number of fibers available on each given span.

$$\sum_{l \in C} \lambda_l^r = d^r, \quad \forall r \in D. \quad (\text{B.5})$$

$$\sum_{l \in C} \varsigma_l^p = \eta^p, \quad \forall p \in P. \quad (\text{B.6})$$

$$\sum_{r \in D} \lambda_l^r \cdot \delta_j^r + \sum_{p \in P: x_j^p=1} \varsigma_l^p \leq F_j, \quad \forall j \in S, \forall l \in C. \quad (\text{B.7})$$

$$\sum_{r \in D, l \in C} \lambda_l^r \cdot \delta_j^r + \sum_{p \in P, l \in C: x_j^p=1} \varsigma_l^p \leq F_j \cdot |C|, \quad \forall j \in S. \quad (\text{B.8})$$

B.4 ILP for Fully Transparent p -Cycle Designs

Equations (B.2)–(B.4), (B.5)–(B.6) and (B.9)–(B.15) comprise the ILP for fully transparent p -cycle designs. This model is very close to the hybrid ILP but constraints (B.3) and (B.7)–(B.8) are now broken in a way that separates working and protection fibers. Accordingly, equations (B.9) and (B.10) counts the number of working fibers and the number of protection fibers in the design. Equations (B.11) and (B.12) ensure that every wavelength is uniquely assigned into a given fiber; (B.11) pertains to working fibers while (B.12) is for protection fibers. Equations (B.13) and (B.14) keep the number of wavelengths assigned in the design under what is available within each fiber. Again, (B.13) relates to working fibers while (B.14) pertains to protection fibers. The objective function is adjusted as well, in order to minimize spare capacity requirements while biasing the selection of p -cycles towards choosing those that involve less (working and protection) fibers in the design.

$$F_{w_j} \geq \frac{w_j}{|C|}, \forall j \in S. \quad (\text{B.9})$$

$$F_{s_j} \geq \frac{s_j}{|C|}, \forall j \in S. \quad (\text{B.10})$$

$$\sum_{r \in D} \lambda_l^r \cdot \delta_j^r \leq F_{w_j}, \forall j \in S, \forall l \in C. \quad (\text{B.11})$$

$$\sum_{p \in P: x_j^p=1} \varsigma_l^p \leq F_{s_j}, \forall j \in S, \forall l \in C. \quad (\text{B.12})$$

$$\sum_{r \in D, l \in C} \lambda_l^r \cdot \delta_j^r \leq F_{w_j} \cdot |C|, \forall j \in S. \quad (\text{B.13})$$

$$\sum_{p \in P, l \in C: x_j^p=1} \varsigma_l^p \leq F_{s_j} \cdot |C|, \forall j \in S. \quad (\text{B.14})$$

$$\text{Minimize } \sum_{j \in S} C_j \cdot s_j + \alpha \cdot \sum_{j \in S} (F_{w_j} + F_{s_j}). \quad (\text{B.15})$$