2007

On Optimal Survivability Design in WDM Optical Networks under Scheduled Traffic Models

Tianjian Li
Wright State University

Follow this and additional works at: http://corescholar.libraries.wright.edu/etd_all
Part of the Computer Engineering Commons, and the Computer Sciences Commons

Repository Citation

This Dissertation is brought to you for free and open access by the Theses and Dissertations at CORE Scholar. It has been accepted for inclusion in Browse all Theses and Dissertations by an authorized administrator of CORE Scholar. For more information, please contact corescholar@www.libraries.wright.edu.
ON OPTIMAL SURVIVABILITY DESIGN IN WDM OPTICAL NETWORKS UNDER SCHEDULED TRAFFIC MODELS

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

By

TIANJIAN LI
M.E., Sichuan University, China, 1999
B.E., Sichuan University, China, 1996

2007
Wright State University
I HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER MY SUPERVISION BY Tianjian Li ENTITLED On Optimal Survivability Design in WDM Optical Networks under Scheduled Traffic Models BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF Doctor of Philosophy.

______________________________
Bin Wang, Ph.D.
Dissertation Director

______________________________
Thomas Sudkamp, Ph.D.
Director, Computer Science and Engineering Ph.D. Program

______________________________
Joseph F. Thomas, Jr., Ph.D.
Dean, School of Graduate Studies

Committee on Final Examination

______________________________
Bin Wang, Ph.D.

______________________________
Thomas Sudkamp, Ph.D.

______________________________
Krishnaprasad Thirunarayan, Ph.D.

______________________________
Xinhui Zhang, Ph.D.

______________________________
Dong Xuan, Ph.D.
ABSTRACT


Wavelength division multiplexing (WDM) optical networks are widely viewed as the most appropriate choice for future Internet backbone with the potential to fulfill the ever-growing demands for bandwidth. WDM divides the enormous bandwidth of an optical fiber into many non-overlapping wavelength channels, each of which may operate at the rate of 10 Gigabit per second or higher. A failure in a network such as a cable cut may result in a tremendous loss of data. Therefore, survivability is a very important issue in WDM optical networks.

The objective of this dissertation is to address the survivability provisioning problem in WDM optical networks under a scheduled traffic model and a sliding scheduled traffic model that we propose. In contrast to the conventional traffic models considered in communication networks such as static traffic model and dynamic random traffic model, the scheduled traffic model and the sliding scheduled traffic model are able to capture the traffic characteristics of applications that require capacity on a time-limited basis. They also give service providers more flexibility in provisioning the requested demands and a better opportunity to optimize the network resources. The survivability provisioning problem is to determine a pair of link-disjoint paths under the link failure model or a pair of SRLG-disjoint paths under the Shared Risk Link Group (SRLG) failure model, one working path and one protection path, for each demand in a given set of traffic demands with the objective of minimizing the total resources used by all traffic demands while 100% restorability is guaranteed against any single failure.

To provision survivable service under the scheduled traffic model, we develop two
sets of integer linear program (ILP) formulations for joint and non-joint optimizations using different protection schemes such as dedicated and shared path based protections. We also design a capacity provision matrix based Iterative Survivable Routing (ISR) algorithm with different demand scheduling policies to solve the survivable routing and wavelength assignment (RWA) problem. In addition, we extend the heuristic algorithm design from dealing with single link failure to single SRLG failure.

The issue of survivability provisioning in WDM optical networks under the sliding scheduled traffic model has never been addressed by the research community. In the dissertation, we carry out the following tasks under this traffic model: (a) development of RWA ILP optimization formulations for dedicated and shared path based protection; (b) design and implementation of efficient heuristic algorithms for shared path based protection. Specifically, in the proposed heuristic algorithm, we introduce a demand time conflict reduction algorithm to minimize the time overlapping among a set of demands by properly placing a demand within its associated time window; and (c) extending the heuristic algorithm design under the single link failure model to the single SRLG failure model.
## CONTENTS

1. Introduction .................................................. 1
   1.1 WDM Optical Networks .................................. 1
       1.1.1 Wavelength Conversion Capabilities .............. 2
       1.1.2 Routing and Wavelength Assignment (RWA) ....... 3
   1.2 Protection and Restoration ............................. 4
   1.3 Traffic Models .......................................... 7
       1.3.1 Conventional Traffic Models ..................... 7
       1.3.2 Scheduled Traffic Model ......................... 9
   1.4 Survivability in Networks with Shared Risk Link Groups ... 10
   1.5 Contributions ......................................... 13
       1.5.1 Survivability Provisioning under Scheduled Traffic Model ... 14
       1.5.2 Sliding Scheduled Traffic Model ................. 16
       1.5.3 Survivability Provisioning under Sliding Scheduled Traffic Model 17
   1.6 Organization of Dissertation ......................... 18

2. Survivability Provisioning under Scheduled Traffic Model .... 19
   2.1 Introduction ........................................... 19
   2.2 Optimization Models .................................. 21
       2.2.1 Joint Optimization Schemes .................... 22
       2.2.2 Two-step Optimization Approach ............... 30
       2.2.3 Numerical and Simulation Results ............. 34
   2.3 Heuristic Algorithm ................................ 38
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3.1</td>
<td>Capacity Provision Matrix based Optimization Model</td>
<td>38</td>
</tr>
<tr>
<td>2.3.2</td>
<td>Determination of Capacity Provision Matrix</td>
<td>41</td>
</tr>
<tr>
<td>2.3.3</td>
<td>Demand Ordering Policies</td>
<td>46</td>
</tr>
<tr>
<td>2.3.4</td>
<td>Iterative Survivable Routing</td>
<td>47</td>
</tr>
<tr>
<td>2.3.5</td>
<td>Numerical and Simulation Results</td>
<td>49</td>
</tr>
<tr>
<td>2.4</td>
<td>Heuristic Algorithm with SRLG</td>
<td>53</td>
</tr>
<tr>
<td>2.4.1</td>
<td>SRLG Failure Model</td>
<td>53</td>
</tr>
<tr>
<td>2.4.2</td>
<td>Iterative Survivable Routing in SRLG Failure Model: ISR-SRLG</td>
<td>58</td>
</tr>
<tr>
<td>2.4.3</td>
<td>Numerical and Simulation Results</td>
<td>61</td>
</tr>
<tr>
<td>2.5</td>
<td>Summary</td>
<td>64</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
<td>68</td>
</tr>
<tr>
<td>3.2</td>
<td>Properties of Sliding Scheduled Traffic Model</td>
<td>69</td>
</tr>
<tr>
<td>3.3</td>
<td>Demand Time Conflict Reduction Algorithm</td>
<td>73</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Strong Edge Detection</td>
<td>73</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Time Conflict Reduction</td>
<td>74</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Weak Edge Removing Policy</td>
<td>75</td>
</tr>
<tr>
<td>3.3.4</td>
<td>Numerical and Simulation Results</td>
<td>77</td>
</tr>
<tr>
<td>3.4</td>
<td>Summary</td>
<td>79</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>82</td>
</tr>
<tr>
<td>4.2</td>
<td>Optimization Models</td>
<td>83</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Joint Optimization Schemes</td>
<td>84</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Numerical and Simulation Results</td>
<td>91</td>
</tr>
<tr>
<td>4.3</td>
<td>Heuristic Algorithm</td>
<td>92</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Two-step Heuristic Approach: ( ISR^+ )</td>
<td>92</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

1.1 Wavelength conversion ........................................ 2
1.2 Different protection and restoration schemes .................. 4
1.3 Protection schemes .............................................. 5
1.4 An illustration of how wavelengths are assigned for the working and protection paths in shared path protection schemes (WP: Working Path; PP: Protection Path) .................. 7
1.5 Time correlation of the example demand set in Table 1.1 ....... 11
1.6 An example network with four ducts and five fibers .......... 12
1.7 Example cases in which duct-layer topology should be taken into consideration ................................. 13

2.1 Three example networks used in [1] .......................... 35
2.2 The NSFNET topology ......................................... 35
2.3 Illustration of capacity provision matrix based optimization model. The numbers besides the links are used to number the links ......... 39
2.4 The weighted interval graph corresponding to the example demand set of Table 1.1 ........................................ 43
2.5 Three demand groups with respect to links l and k ........... 43
2.6 Iterative Survivable Routing algorithm ........................ 48
2.7 Single demand routing algorithm for finding a working path of demand r .................................................. 50
2.8 Total number of wavelength-links versus number of demands .... 53
2.9 Illustration of capacity provision matrix based optimization model with SRLGs. The numbers besides the links (but not in the parentheses) are used to number the links while the numbers in the parentheses indicate the corresponding SRLG that the link belong to. 56

2.10 Three demand groups with respect to link \( l \) and links in \( S(f) \). 57

2.11 Iterative Survivable Routing algorithm under the single SRLG failure model. 59

2.12 Single demand routing algorithm for finding a working path under the single SRLG failure model. 61

2.13 Comparison of SRLG size: Total number of wavelength-links versus number of demands under weak time correlation. 63

2.14 Comparison of SRLG size: Total number of wavelength-links versus number of demands under medium time correlation. 63

2.15 Comparison of SRLG size: Total number of wavelength-links versus number of demands under strong time correlation. 64

2.16 Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links used in 2-SRLG NSFNET network. 65

3.1 A sliding scheduled traffic demand. 69

3.2 Interval graph representation of the example demand set in Table 1.1. 70

3.3 An example that shows how time conflict reduction algorithm works. Filled vertices are tight vertices. Empty vertices are loose vertices. Thick edges are strong edges. Thin edges are weak edges. 74

3.4 Pseudo code for the demand time conflict reduction algorithm. 75

3.5 Demand time conflict reduction: Percentage of demand time conflicts resolved for weak time window correlation. 78

3.6 Demand time conflict reduction: Percentage of demand time conflicts resolved for medium time window correlation. 79
3.7 Demand time conflict reduction: Percentage of demand time conflicts resolved for strong time window correlation. ........................................ 80

3.8 Demand time conflict reduction with earliest setup demand first policy: Percentage of demand time conflicts resolved for weak, medium and strong time window correlation. The last figure shows the result for demand sets with uniformly mixed demand holding time with respect to the time window length. ........................................ 81

4.1 Total number of wavelength-links versus number of demands under weak time window correlation. ........................................ 95

4.2 Total number of wavelength-links versus number of demands under medium time window correlation. ........................................ 96

4.3 Total number of wavelength-links versus number of demands under strong time window correlation. ........................................ 97

4.4 Total number of wavelength-links versus number of demands under weak, medium and strong time window correlation. The last figure shows the result for demand sets with uniformly mixed demand holding time with respect to the time window length. ........................................ 98

4.5 Comparison of SRLG size: Total number of wavelength-links versus number of demands under weak time correlation (part 1). ............ 101

4.6 Comparison of SRLG size: Total number of wavelength-links versus number of demands under weak time correlation (part 2). ............ 102

4.7 Comparison of SRLG size: Total number of wavelength-links versus number of demands under medium time correlation (part 1). ............ 103

4.8 Comparison of SRLG size: Total number of wavelength-links versus number of demands under medium time correlation (part 2). ............ 104

4.9 Comparison of SRLG size: Total number of wavelength-links versus number of demands under strong time correlation (part 1). ............ 105
4.10 Comparison of SRLG size: Total number of wavelength-links versus number of demands under strong time correlation (part 2). . . . . . . 106

4.11 Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links versus number of demands under weak time window correlation in 2-SRLG NSFNET network. . . . . . . . . 107

4.12 Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links versus number of demands under medium time window correlation in 2-SRLG NSFNET network. . . . . . . . . 108

4.13 Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links versus number of demands under strong time window correlation in 2-SRLG NSFNET network. . . . . . . . . 109
LIST OF TABLES

1.1  An example of a scheduled demand set. 

2.1  Scenario Information. 

2.2  Total number of wavelength-links used. (∗ indicates the optimal solution found and W, M, S represent weak, medium and strong time correlation, respectively.) 

2.3  Total number of wavelength-links used. * indicates the optimal solution found and W, M, S represent weak, medium and strong time correlation, respectively. τ is the time correlation of a demand set. 

2.4  Computational time. × indicates the 4-hour time limit. h, m and s represent hour(s), minute(s) and second(s), respectively. τ is the time correlation of a demand set. 

4.1  Total number of wavelength-links used in protection scheme DP-SS. (∗ indicates the optimal solutions found and W, M, S represent weak, medium and strong time correlation, respectively.) 

4.2  Total number of wavelength-links used in protection scheme SP-SS. (∗ indicates the optimal solutions found and W, M, S represent weak, medium and strong time correlation, respectively.) 

4.3  Total number of wavelength-links used in protection scheme SP-SS in sliding heuristic approach ISR.
ACKNOWLEDGMENTS

First of all, I would like to thank my advisor Dr. Bin Wang for introducing me to the field of optical networking. I feel inspired by his knowledge, hard work, dedication to excellence and careful attention to details. This dissertation would not be possible without his guidance, encouragement and support. The extensive discussion with him on broad topics and deliberate revisions on my draft papers during the past four and half years have helped me stay on the right track toward finishing this work.

I would also like to thank the other members of my dissertation committee - Dr. Thomas Sudkamp, Dr. Krishnaprasad Thirunarayan, Dr. Xinhui Zhang and Dr. Dong Xuan for their precious time in reviewing the manuscript and their valuable suggestions. I am also grateful to them for their encouragement and support on my research and studies.

I am very fortunate to work with a group of students in the Advanced Networking Research Lab at Wright State, including Xubin Luo, Yuqi Fan, Ashish Deshmukh, Abdur Billah, Michael Greene, Sowjanya Chava, Nanda Lella, Towsif Mannan and Tao Zhang. I had a great time working with them.

I owe a lot more than a thank to my parents, parents-in-law, elder brother and my wife. They have been always encouraging and supporting. Without their love and inspiration, I would not have made through this most important stage of my life.

Finally, I have been supported financially by the Dayton Area Graduate Studies Institute (DAGSI) and the Department of Energy Early Career Award during the past three and half years. I would like to express my gratitude to DAGSI and DoE for the generous support.
1. INTRODUCTION

The popularity of the Internet has resulted in an exponential growth in bandwidth demands. Wavelength division multiplexing (WDM) divides the enormous bandwidth of an optical fiber into many non-overlapping wavelength channels. WDM optical networks are able to meet the rapid growth of bandwidth demands and are considered to be the most appropriate choice for future Internet backbone. Currently, dense WDM (DWDM) technologies can achieve up to 320 wavelengths per fiber with each wavelength carrying 10 Gbps, resulting in a total transmission capacity of up to 3.2 Tbps [2]. It is expected that each wavelength channel can carry 40 Gbps or more in the near future. Since a fiber is capable of carrying multiple wavelengths simultaneously, each of which may operate at the rate of 10 Gigabit per second or higher, a failure in a network such as a cable cut may result in a tremendous loss of data. Therefore, survivability is a very important issue in WDM optical networks and has attracted much recent research.

1.1 WDM Optical Networks

In the dissertation, we consider all-optical networks wherein signals are processed optically from end to end, i.e., without any opto-electronic (O/E) or electro-optical (E/O) conversion at intermediate switching equipment. An end-to-end all optical connection is called a lightpath.
1.1.1 Wavelength Conversion Capabilities

In a WDM optical network, an optical switch or cross-connect (OXC) is wavelength conversion capable if it can switch a signal from an input port on one wavelength channel to an output port on another wavelength channel. Wavelength-routed networks with this capacity are referred to as wavelength-convertible networks [3]. Different levels of wavelength conversion capability are possible. An optical switch is said to have full conversion capability if it can convert any given wavelength channel to any other wavelength. A wavelength-convertible network that supports full wavelength conversion at all nodes is functionally equivalent to a circuit-switched network, i.e., connection requests are blocked only when there is no available capacity on the path. Wavelength-convertible networks may have limited wavelength conversion capabilities. For example, a network is said to have limited conversion if all the switches in it can only convert any given wavelength channel to a limited range of wavelengths; it is said to be sparse conversion capable if only a limited number of switches in the network have full conversion capability. When wavelength conversion is not available, a lightpath must use the same wavelength on all the links traversed in a WDM optical network. This requirement is known as the wavelength continuity constraint. Wavelength-routed networks with this constraint are referred to as wavelength-continuous networks. Fig. 1.1 illustrates the differences for a single input and single output port situation; the case for multiple ports is more complicated but similar.

\[ \begin{align*}
\lambda_1 & \rightarrow \lambda_1 \\
\lambda_2 & \rightarrow \lambda_2 \\
\lambda_3 & \rightarrow \lambda_3 \\
\end{align*} \quad \begin{align*}
\lambda_1 & \rightarrow \lambda_1 \\
\lambda_2 & \rightarrow \lambda_2 \\
\lambda_3 & \rightarrow \lambda_3 \\
\end{align*} \quad \begin{align*}
\lambda_1 & \rightarrow \lambda_1 \\
\lambda_2 & \rightarrow \lambda_2 \\
\lambda_3 & \rightarrow \lambda_3 \\
\end{align*}

(a) No conversion \quad (b) Limited conversion \quad (c) Full conversion

*Fig. 1.1: Wavelength conversion.*
Although previous research [3, 4] has shown that wavelength conversion enables more efficient resource utilization and may reduce the lightpath blocking probability significantly by resolving the wavelength conflicts of lightpath routing, wavelength converters should not be used arbitrarily due to their high costs and possible signal quality degradation for some converter types. It has been demonstrated that a relatively small number of converters is sufficient for a certain level of performance [5–13]. In this work, however, we assume full wavelength conversion in all-optical wavelength-routed networks. This assumption simplifies the problems investigated in the work and allows us to concentrate on their essential characters.

1.1.2 Routing and Wavelength Assignment (RWA)

A unique feature of optical WDM networks is the tight coupling between routing and wavelength assignment. A lightpath is established in a network by selecting a route of physical links between the source and destination nodes, and reserving a particular wavelength on each of these links for the lightpath. In establishing an optical connection, therefore, we must deal with both routing (selecting a suitable route) and wavelength assignment (allocating an available wavelength for the connection). The resulting problem is referred to as the routing and wavelength assignment (RWA) problem [14]. It is significantly more difficult than the routing problem in electronic networks. In fact, most RWA problems have been proved to be \textit{NP}-complete. For instance, the problem of setting up lightpaths for all the connection demands in a given demand set is known as the static lightpath establishment (SLE) problem [14]. The objective of the problem is to minimize network resources such as the number of wavelengths or the number of wavelength-links in the network. The work [14] showed that the SLE problem is a \textit{NP}-complete problem.
1.2 Protection and Restoration

Various approaches to provisioning survivability exist for both non-WDM and WDM optical networks [15]. Fig. 1.2 illustrates different approaches. These approaches can generally be divided into protection (a.k.a. pre-designed approaches, proactive approaches) and restoration (a.k.a. reactive approaches) [16]. Specifically, protection refers to techniques that use preassigned capacity to ensure survivability while techniques that re-route affected traffic after a failure occurrence using available capacity is referred to as restoration [16]. In general, protection based approaches offer faster recovery times while restoration based approaches may be more resource efficient. Various fault-resilient schemes can be designed at IP and/or WDM layers to protect users’ traffic from disruptions due to failures. WDM layer survivability is desirable due to its many advantages: speed, simplicity, effectiveness, and transparency [15].

![Protection/Restoration Schemes](image)

**Fig. 1.2:** Different protection and restoration schemes.

Protection based approaches can be further divided into path based and link based schemes, respectively. In path based protection schemes, upon a link failure, the source and the destination of each connection affected by the failure switch to its corresponding protection wavelength path that is routed on a fiber-disjoint path from
the affected connection, as shown in Fig. 1.3(a). Link based protection schemes re-route (or loop back) traffic around the failed link and involve only the nodes adjacent to the link failure (i.e., the end nodes of the failed link are responsible for recovery), as shown in Fig. 1.3(b). On the other hand, path based protection schemes need a mechanism to notify the source and the destination of the affected connection of the failure. However, path based protection schemes may be more resource efficient and usually offer a shorter end-to-end recovery route than link based protection techniques [17, 18]. We therefore consider path based protection in this work.

![Fig. 1.3: Protection schemes.](image)

Based on whether backup network resource sharing is allowed or not, path based protection schemes can be classified into dedicated path based protection approaches and shared path based protection approaches. In the former case, the resources on the links of a protection path are reserved for a given working connection. Two protection paths sharing some common fiber links must use different wavelengths even if their corresponding working paths are link-disjoint. In the latter case, on the other hand, resource sharing among protection paths is allowed and multiple protection paths can go through common fiber links as long as their working and protection paths satisfy certain constraints, e.g., their working paths are link-disjoint. For example, we need to establish three connections in an example network shown in Fig. 1.4 with the objective of minimizing the total wavelength-links used in the network by employing
shared path protection scheme. We assume the connection requests are processed sequentially and the wavelength continuity constraint is required in the network. In Fig. 1.4(a), the first connection (0, 1) has been established in the network with a working path (0 → 1) and a protection path (0 → 4 → 1). Both paths use wavelength 0 on all the links they traverse. When the second connection request (0, 2) arrives, paths (0 → 1 → 2) and (0 → 4 → 5 → 2) are chosen to be the working path and the protection path, respectively. The two working paths and the two protection paths (Fig. 1.4(b)) use a common link (0, 1) and link (0, 4), respectively. Since any two link-joint working paths must use different wavelengths, wavelength 1 on link (0, 1) and link (1, 2) are assigned to working path $WP_2$. We use $(i, j : k)$ to refer to the $k$-th wavelength-link on link $(i, j)$. Since these two working paths are link-joint, protection path $PP_2$ and protection path $PP_1$ cannot share the same wavelength-link (0, 4 : 0). Instead $PP_2$ has to use wavelength 1 in order to guarantee 100% restorability in case that link (0, 1) fails. Suppose a third connection request (0, 3) arrives after these two connections have been established in the network. If we jointly solve the RWA problem given this request, the optimal solution is to choose path (0 → 6 → 7 → 3) as the working path and (0 → 4 → 5 → 3) as the protection path. The total number of additional wavelength-links to accommodate this connection request with shared path protection is 4 which is the minimum among all feasible solutions. Fig. 1.4(c) shows the network state after the third request has been established. In the figure, protection path $PP_3$ shares wavelength-links (0, 4 : 1) and (4, 5 : 1) with protection path $PP_2$ and wavelength-link (5, 3 : 1) is the only additional wavelength-link needed by the new protection path. Therefore, overall 4 additional wavelength-links are needed to accommodate the third request. By employing shared protection, therefore, it is possible to utilize network resources more efficiently while still achieving 100% restorability against single failure. The recovery time for shared path protection schemes may be longer; but the overall resource utilization is much better compared
with that of dedicated protection schemes [19,20]. In this work, we will primarily focus on shared path protection.

![Diagram of shared path protection](image)

*Fig. 1.4: An illustration of how wavelengths are assigned for the working and protection paths in shared path protection schemes (WP: Working Path; PP: Protection Path).*

### 1.3 Traffic Models

#### 1.3.1 Conventional Traffic Models

A great deal of research has been conducted on survivability provisioning in WDM optical networks. Previous work has considered several types of traffic models, e.g., static traffic, dynamic random traffic, admissible set, and incremental traffic. In the static traffic model, all demands are known in advance and do not change over time. For instance, a client company may request virtual private connectivity among different company sites from a service provider. The objective is typically to minimize the network resources, e.g., the number of wavelengths, converters, etc, or to maximize throughput given a resource constraint [21]. This model does not allow dynamic call
setup and tear-down. In the dynamic random traffic model, a demand is assumed to arrive at a random time and last for a random amount of time. Usually statistical models are used. These models assume certain arrival statistics (e.g., Poisson process) and holding time (e.g., exponential distribution) for demands, as well as a certain traffic distribution (e.g., uniform traffic). The design objective is typically to minimize the call blocking probability, or to analytically model the call blocking probability under various assumptions [22–28]. In the admissible set model, the objective is to design networks to accommodate any traffic matrix from an admissible set. The set of traffic matrices may be characterized by the maximum link load in the network [29], or by actual device limitations in the network [30–32], e.g., the numbers of tunable transmitters and tunable receivers at each end node (i.e., a node that sources and/or sinks traffic sessions). A new session is said to be allowable if its arrival results in a traffic matrix which is still in the set of admissible traffic. The goal is to minimize the number of wavelengths used. The work in [30–33] mainly targeted at simple network topologies (i.e., ring and torus). The work reported in [34] considered time-variant offered traffic in the form of a set of traffic matrices at different instants for off-line configuration so as to accommodate such time-varying traffic. The work in [35] also used a set of traffic matrices to design and dimension a WDM mesh network to groom dynamically varying traffic. In the incremental model [36, 37], traffic demands arrive sequentially. Lightpaths are established for each demand, and remain in the network indefinitely. The work in [38] on multi-period network planning was based on an incremental traffic model and conducted network planning across several years to produce incrementally a network capable of carrying all traffic predicted up to the end of the planning horizon.

The literature on survivability provisioning in optical networks under conventional traffic models is abundant (see [39–41] and references therein). For example, the work in [1, 11, 41–45] proposed several joint working and protection paths planning
approaches in survivable WDM networks. The corresponding optimization problems have been proved to be \( NP \)-complete. Some recent work [11, 19, 46–48] considered dynamic routing and wavelength assignment of lightpaths with protection requirements.

### 1.3.2 Scheduled Traffic Model

While the conventional traffic models are valid and useful in many circumstances, they are not able to capture the traffic characteristics of applications that require capacity during specific time intervals or circuit leasing on a short term basis. For instance, a client company may request some scheduled demands for bandwidth from a service provider to satisfy its communication requirements at a specific time, e.g., between headquarters and production centers during office hours or between data centers during the night when backup of databases is performed and so on. Other examples include many US Department of Energy large-scale science applications (e.g., applications in high energy physics, climate data and computations, astrophysics, etc) that must deliver, at scheduled time durations, hundreds of Gbps throughput between two applications in near future and several Tbps within the next decade, ranging from cooperative remote visualization of massive archival data through the distribution of large amount of simulation data, to the interactive evolution of computations through computational steering [49]. These applications require provisioning of scheduled dedicated channels or bandwidth pipes at a specific time with certain duration. These scheduled bandwidth demands [50] are dynamic in nature. They are not static since the demands only last during the specified intervals. They are not entirely random either. In real networks, a mix of static, dynamic and scheduled traffic is expected. The case studied in this dissertation with scheduled traffic only, though somewhat extreme, is justified to study the extent of survivability performance gain under the scheduled traffic model.
In this dissertation, we consider a scheduled traffic model in which a set of scheduled traffic demands, $D$, is given, each demand of which is represented by a tuple $r = (s, d, n, \alpha, \beta)$, where $s$ and $d$ are the source and destination nodes of the demand, $n$ is the number of requested lightpaths, $\alpha$ and $\beta$ are the setup and teardown times of the demand, respectively. We use $D$ to denote the total number of demands. The scheduled traffic model is different from the static and dynamic random traffic models generally assumed in the literature. In the static traffic model, all demands are known in advance and do not change over time, while the dynamic random traffic assumes that a demand arrives at a random time, the inter-arrival time and holding time of demands are random or conform to some probability distribution. The scheduled traffic model is more deterministic, and the time dimension of demands is explicitly considered since many demands for bandwidth in ultra high-speed networks will be short-lived in contrast to $7 \times 24$ operations. The model is also dynamic since demands only last during the specified time intervals.

Given a set of scheduled demands, some demands may not overlap in time. For example, Table 1.1 shows an example of a scheduled demand set which includes 7 demands. The time correlation of these demands is shown in Fig. 1.5. It is easy to find that demands $r_1$ and $r_4$ are time disjoint. Since the network resources used by demand $r_1$ have been released when demand $r_4$ is scheduled, the resources can be completely reused by demand $r_4$. This motivates us to take into account the time disjointness (if any) among demands along both working and protection paths in addition to optimizing the spatial network resource sharing based on backup resource sharing, to achieve a higher degree of overall network resource shareability.

1.4 Survivability in Networks with Shared Risk Link Groups

A network failure may be caused by either a link failure or a node failure. Most modern node devices have built-in redundancy which greatly improves their reliability.
Therefore link failure is more of a concern than node failure, and we only consider link failure in this work. In path protection schemes, as we mentioned before, the working path and the protection path of a same connection must be link-disjoint so that the network is survivable under single link failure. However, in a network where a single factor can cause more than one link failure, the two link-disjoint lightpaths found may still fail simultaneously. In practice, for instance, multiple fibers\(^1\) are bundled into the same underground duct\(^2\), or span. A fiber cut usually occurs due to a duct cut during construction or destructive natural events such as earthquakes. When a duct is cut, normally all the fibers in the duct fail at the same time. Hence, a network survivable to a single fiber failure is not necessarily survivable in duct failure scenarios. As an example, a network with four nodes, four ducts and five fibers is shown in Fig. 1.6(a), and its corresponding link-layer topology is shown in Fig. 1.6(b). If we compute two link-disjoint paths from node 0 to node 1 purely on

\(^{1}\) In this work, the term “fiber” refers to a bidirectional fiber link (or a pair of fibers, one in each direction) while the term “link” refers to an unidirectional fiber link.

\(^{2}\) A duct is a bidirectional physical pipe between two end nodes.

---

<table>
<thead>
<tr>
<th>Demands</th>
<th>s</th>
<th>d</th>
<th>n</th>
<th>α</th>
<th>β</th>
</tr>
</thead>
<tbody>
<tr>
<td>r_1</td>
<td>2</td>
<td>9</td>
<td>1</td>
<td>05:00</td>
<td>09:20</td>
</tr>
<tr>
<td>r_2</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>07:00</td>
<td>12:40</td>
</tr>
<tr>
<td>r_3</td>
<td>1</td>
<td>6</td>
<td>2</td>
<td>08:00</td>
<td>14:00</td>
</tr>
<tr>
<td>r_4</td>
<td>4</td>
<td>8</td>
<td>2</td>
<td>11:00</td>
<td>16:00</td>
</tr>
<tr>
<td>r_5</td>
<td>7</td>
<td>2</td>
<td>3</td>
<td>12:00</td>
<td>14:50</td>
</tr>
<tr>
<td>r_6</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>17:00</td>
<td>21:00</td>
</tr>
<tr>
<td>r_7</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>18:00</td>
<td>21:00</td>
</tr>
</tbody>
</table>

Tab. 1.1: An example of a scheduled demand set.
the link-layer topology, we might get the working path as \((0 \rightarrow 1)\) and the protection path as \((0 \rightarrow 3 \rightarrow 1)\), as shown in Fig. 1.7(a). However, note that link \((0,1)\) and link \((0,3)\) pass through the same duct \((0,1)\), and, hence, both may fail due to a failure on duct \((0,1)\). Therefore we must find a pair of duct-disjoint paths as the working path and the protection path for the connection request. For instance, paths \((0 \rightarrow 1)\) and \((0 \rightarrow 2 \rightarrow 3 \rightarrow 1)\) is a feasible solution.

We must also consider the duct-layer topology when we employ shared path protection schemes. For example, there are two connections being set up in the network shown in Fig. 1.6, one from node 0 to node 1, and the other from node 0 to node 3. Fig. 1.7(b) shows their working paths and protection paths established in the network. In the shared path protection scheme, if we only look at the link-layer topology, we may allow the two protection paths to share the same wavelengths on links \((0,2)\) and \((2,3)\) because their working paths are link-disjoint. However, the two working paths actually go through the same duct \((0,1)\) and thus they may fail at the same time. Consequently, we should use different wavelengths for the two protection paths on links \((0,2)\) and \((2,3)\).

More generally, transport network carriers use the notation of Shared Risk Link Group (SRLG) \([51]\), which associates all the links with a failure, to describe the type of network phenomenon mentioned before. Obviously the fiber links in the same
duct belong to the same SRLG because they all share the same risk of a duct cut. To guarantee 100% restorability against any single SRLG failure in a WDM optical network, therefore, the protection path of a given connection should not share any SRLG with the working path of the same connection. We call these two paths SRLG-disjoint. For some special SRLG configurations, such as forks and express links, there exist algorithms with polynomial time complexity to find two SRLG-disjoint paths [52, 53]. If the configurations are arbitrary, it has been proved that the problem of finding two SRLG-disjoint paths is \( \mathcal{NP} \)-complete [54, 55].

1.5 Contributions

Due to the high capacity of a WDM optical network, a failure in the network such as a fiber cut may result in a tremendous loss of data. Therefore, survivability provisioning is one of the most important issues in WDM optical networks. During the past decade, a great deal of research work has been conducted in this area. However, most previous work concentrated on conventional traffic models, especially on static traffic model and dynamic random traffic model, as described in Chapter 1.3. Only some recent work including ours paid attention to the survivability problems under the scheduled

---

Fig. 1.7: Example cases in which duct-layer topology should be taken into consideration.
traffic model.

The central theme of this dissertation is to develop optimization models, in terms of integer linear program (ILP) formulations, and design efficient heuristic algorithms for survivability provisioning in WDM optical networks under the scheduled traffic model and the sliding scheduled traffic model that we propose in this dissertation. The overall objective is, given a set of scheduled traffic demands $D$, to find one working path and one protection path that are link-disjoint (in the link failure model) or SRLG-disjoint (in the SRLG failure model) for each of the demands in the set while minimizing the total resources used by working paths and protection paths of all traffic demands in $D$. In our approaches, network resource reuse is maximally exploited in both space and time while 100% restorability is guaranteed against any single link failure (in the link failure model) or any single SRLG failure (in the SRLG failure model).

1.5.1 Survivability Provisioning under Scheduled Traffic Model

Previous work that considers scheduled traffic model is very limited. In the work of Kuri et al. [50, 59], a scheduled lightpath demand model was proposed. The routing and wavelength assignment problem is solved using a branch & bound algorithm and a tabu search algorithm. The issue of diverse routing of scheduled lightpath demands was addressed in Kuri’s another work [60]. They formulated it into an optimization model, which is basically a two-step optimization approach, and proposed a simulated annealing based algorithm to find approximate solutions to the optimization problem. The work of Tornatore et al. [61] exploits the connection-holding-time information to dynamically provision shared-path-protected connections using heuristic algorithms. The work of Saradhi [62] considered the provisioning of fault-tolerant scheduled lightpath demands based on a two-step optimization that uses a set of pre-computed routes for working and protection paths. However, their approaches appear to be
flawed. We listed some problems of it in details in our work [63]. In [63], furthermore, we provided a set of correct joint RWA problem formulations that enable the maximum resource sharing in both space and time (i.e., use backup resource sharing and resource sharing exploiting time disjointness among demands). We also used a two-step optimization approach which divides the joint RWA problems into a routing subproblem and wavelength assignment subproblems and then solves them individually. In our recent work [64], we proposed an efficient capacity provision matrix based heuristic algorithm with different demand ordering policies to solve the survivable provisioning problem under the scheduled traffic model. The details of the work reported in [63] and [64] will be explained in Chapter 2.

Our contributions for survivability provisioning in WDM optical networks under the scheduled traffic model include optimization model development and heuristic algorithm design.

- **Optimization Models:** We develop joint RWA ILP formulations for dedicated and shared protection schemes in survivable WDM optical mesh networks under both the conventional static traffic model and the scheduled traffic model, respectively. The formulations under the former model are used for performance evaluation of the schemes under the latter model. To solve large problems, we divide the joint RWA problem into a routing subproblem and a wavelength assignment subproblem. For the routing subproblem, we use an ILP to pre-compute a pair of link-disjoint routes for each of the demands in the traffic demand set $D$ as its working path and protection path. For the wavelength assignment subproblems, the pair of routes for each demand $r$ in $D$ is assigned proper wavelengths such that the total network resources (i.e., the number of wavelength-links) used by all the demands are minimized.

- **Heuristic Algorithms:** We also propose an efficient heuristic algorithm employing shared path protection scheme to solve the survivable routing problem un-
der the scheduled traffic model. The proposed approach is based on an iterative survivable routing (ISR) scheme that utilizes a capacity provision matrix and processes demands sequentially using different demand ordering policies. Our simulation results indicate that the proposed ISR algorithm is extremely time efficient while achieving excellent performance in terms of total network resources used. The impact of demand ordering policies on the ISR algorithm is also studied.

- **Heuristic Algorithms with SRLG:** We extend the heuristic algorithm design from dealing with single link failure to single SRLG failure. The extended heuristic algorithm ISR-SRLG is able to solve the survivable routing problem in WDM optical networks in which 100% restorability is guaranteed against any single SRLG failure under the scheduled traffic model. We consider both localized SRLGs and non-localized SRLGs, as well as SRLG sets with different sizes.

1.5.2 **Sliding Scheduled Traffic Model**

In this dissertation, we propose a sliding scheduled traffic model, which is more general than the scheduled traffic model and gives service providers more flexibility in provisioning the requested demands and a better opportunity to optimize the network resources. In addition to the introduction of the sliding scheduled traffic model, our contributions with respect to this model are as follows:

- **Model Properties:** We derive a set of properties of the sliding scheduled traffic model. These properties will be used in the optimization model development and heuristic algorithm design under the sliding scheduled traffic model.

- **Demand Time Conflict Reduction Algorithm:** We design a demand time conflict reduction algorithm to maximize temporal resource reuse in the sliding scheduled traffic model. The algorithm reduces the time overlapping among a set
of demands by properly placing a demand within its associated time window. Our simulation results show that the proposed demand time conflict reduction algorithm can resolve well over 50% of time conflicts.

1.5.3 Survivability Provisioning under Sliding Scheduled Traffic Model

In the work of [56–58], we designed some efficient heuristic algorithms to solve the RWA problem for a single demand or a given demand set, respectively, under the sliding scheduled traffic model, but we only find a single path, i.e., a working path, for each of the demands. In other words, survivability is not considered in all the work. As a matter of fact, there has been no published work that investigates the issue of survivability provisioning in WDM optical networks under the sliding scheduled traffic model. The following summarizes our major contributions in this area:

- **Optimization Models:** We develop joint RWA ILP formulations for dedicated and shared protection schemes in survivable WDM optical mesh networks under the sliding scheduled traffic model. These optimization models maximally exploit the network resource reuse in both space and time under the sliding scheduled traffic model.

- **Heuristic Algorithms:** We develop an efficient two-step approach $ISR^+$ to solve the survivable routing problem under the sliding scheduled traffic model. In the first step of the approach, we use the demand time conflict reduction algorithm to maximize temporal resource reuse. In the second step, we use the ISR algorithm to find routes and assign wavelengths to the demands whose starting times and ending times have been determined by the demand time conflict reduction algorithm.

- **Heuristic Algorithms with SRLG:** We extend the heuristic algorithm design from the single link failure model to the single SRLG failure model. The extended
heuristic algorithm $ISR^+$-SRLG is able to solve the survivable routing problem in WDM optical networks in which 100% restorability is guaranteed against any single SRLG failure under the sliding scheduled traffic model. Similar to ISR-SRLG, we consider both localized SRLGs and non-localized SRLGs, as well as SRLG sets with different sizes.

1.6 Organization of Dissertation

The rest of the dissertation is organized as follows. Chapter 2 presents the development of survivability optimization models and the design of efficient heuristic algorithms to provisioning survivability in WDM optical networks under the scheduled traffic model. The heuristic algorithm design is then extended from the single link failure model to the single SRLG failure model. Numerical and simulation results of all the work are also given in this chapter. Chapter 3 proposes a general scheduled traffic model – the sliding scheduled traffic model. A set of properties of this model are derived and, based on these properties, a demand time conflict reduction algorithm is introduced to minimize the time overlapping among a set of demands by properly placing a demand within its associated time window. Chapter 4 presents ILP formulations, a heuristic algorithm and its SRLG extension for survivability provisioning under the sliding scheduled traffic model. Finally, this dissertation is concluded in Chapter 5 which also presents some future research topics.
2. SURVIVABILITY PROVISIONING UNDER SCHEDULED TRAFFIC MODEL

2.1 Introduction

During the past decade, a great deal of research work has been conducted in survivability provisioning, which has been recognized as one of the most important issues in WDM optical networks. However, most previous work concentrated on conventional traffic models, especially on static traffic model and dynamic random traffic model, as described in Section 1.3. While the conventional traffic models are valid and useful in many circumstances, they are not able to capture the traffic characteristics of applications that require capacity during specific time intervals or circuit leasing on a short term basis. This motivates us to study the survivability provisioning problem under the scheduled traffic model.

Previous work that considers scheduled traffic model is very limited and some of them appear to be flawed, as mentioned in Section 1.5.1. In this study, we conducted comprehensive research on survivability provisioning in WDM optical networks under the scheduled traffic model. We also studied survivability provisioning in WDM networks with Shared Risk Link Group (SRLG), which is more general than the link failure model. Our research work includes optimization model development and heuristic algorithm design. The overall objective is, given a set of scheduled traffic demands $\mathcal{D}$, to find one working path and one protection path that are link-disjoint (in the link failure model) or SRLG-disjoint (in the SRLG failure model) for each of the demands in the set while minimizing the total resources used by working paths and protection paths of all traffic demands in $\mathcal{D}$. In our approaches, network
resource reuse is maximally exploited in both space and time while 100% restorability is guaranteed against any single link failure (in the link failure model) or any single SRLG failure (in the SRLG failure model).

We first develop joint RWA ILP formulations for dedicated and shared protection schemes in survivable WDM optical mesh networks under both the conventional static traffic model and the scheduled traffic model, respectively. The formulations under the former model are used for performance evaluation of the schemes under the latter model. To solve large problems, we separate the joint RWA problem into a routing subproblem and a wavelength assignment subproblem. For the routing subproblem, we use an ILP to pre-compute a pair of link-disjoint routes for each of the demands in the traffic demand set \( D \) as its working path and protection path. For the wavelength assignment subproblems, the pair of routes for each demand \( r \) in \( D \) is assigned proper wavelengths such that the total network resources (i.e., the number of wavelength-links) used by all the demands are minimized. We then propose an efficient heuristic algorithm to solve the survivable routing problem under the scheduled traffic model by processing demands sequentially. We consider different demand ordering policies to investigate the impact of the demand processing order. Finally, we extend the heuristic algorithm design from the single link failure model to the single SRLG failure model. The extended heuristic algorithm is expected to solve the survivable routing problem in WDM optical networks in which 100% restorability is guaranteed against any single SRLG failure under the scheduled traffic model.

The rest of the chapter is organized as follows. Section 2.2 presents joint RWA ILP formulations and two-step RWA ILP formulations for survivability provisioning under the scheduled traffic model. Section 2.3 describes an efficient heuristic algorithm, called Iterative Survivable Routing, which processes demands sequentially. The demand processing order is determined in different demand ordering policies. In Section 2.4, we present the SRLG failure model and extend the Iterative Survivable
Routing algorithm to solve the survivability provisioning problem under the SRLG failure model and the scheduled traffic model. Section 2.5 summarizes the chapter.

2.2 Optimization Models

In this section, we develop integer linear program (ILP) formulations for dedicated and shared protection schemes in survivable WDM optical mesh networks under both the conventional static traffic model and the scheduled traffic model, respectively. The formulations under the former model are used for performance evaluation of the schemes under the latter model. Overall, four architectures are investigated and they are abbreviated as follows:

- **DP**: dedicated path protection under the conventional static traffic model (connection holding time unaware);
- **SP**: shared path protection under the conventional static traffic model (connection holding time unaware);
- **DP-S**: dedicated path protection under the scheduled traffic model (connection holding time aware);
- **SP-S**: shared path protection under the scheduled traffic model (connection holding time aware).

To obtain the optimal solutions for survivable service provisioning problems, routing and wavelength assignment (RWA) for all the demands in a given traffic demand set $D$ should be conducted jointly in the ILPs. We first develop four joint RWA ILP formulations, ILP1, ILP2, ILP3 and ILP4, for each of the four architectures DP, SP, DP-S and SP-S, respectively. The formulations under the first two architectures (ILP1 and ILP2, as well as ILP6 discussed below) serve as the references for performance evaluation of those under the other two architectures. The objective of all the ILPs is to minimize the total number of wavelength-links used.
To solve large problems, however, we found the joint formulations $\text{ILP2, ILP3}$ and $\text{ILP4}$ are too complex in terms of the number of variables and the size of search space. To reduce the computational complexity, we propose a two-step optimization approach for each of the three architectures $\text{SP, DP-S and SP-S}$. Specifically, we partition the RWA problems into a routing subproblem and wavelength assignment subproblems, as did in [1]. For the routing subproblem, we use an ILP ($\text{ILP5}$) to pre-compute a pair of link-disjoint routes for each of the demands in $\mathcal{D}$ as its working path and protection path. For the wavelength assignment subproblems, we present three ILPs ($\text{ILP6, ILP7 and ILP8}$) for the three protection schemes $\text{SP, DP-S and SP-S}$, respectively. In the wavelength assignment subproblems, the pair of routes for each demand $r$ in the traffic demand set $\mathcal{D}$ will be assigned proper wavelengths such that the total network resources (i.e., the number of wavelength-links) used by all the demands are minimized. We assume that network resources are sufficient to satisfy the entire traffic demand set, and that the network has full wavelength conversion capability. The wavelength assignment problem is important to maximally exploit the resource sharing in the time domain enabled by the given demand holding time even though the network has full wavelength conversion capability.

2.2.1 Joint Optimization Schemes

**ILP1 – ILP for DP**

This ILP is developed for joint RWA optimization using dedicated path protection under the static traffic model (i.e., holding time unaware). In this optimization model, all the demands are assumed unchanged over time and no network resource sharing is allowed among any protection paths.

(1) The following are given as program inputs:

- $\mathcal{N}$: the set of nodes in the network.
- $\mathcal{L}$: the set of links in the network.
• $\mathcal{K}$: the set of wavelengths on each link.

• $\mathcal{D}$: the set of traffic demands; For each demand $r \in \mathcal{D}$, $s_r$, $d_r$ and $n_r$ are the source node, destination node, and the number of requested lightpaths of demand $r$, respectively.

(2) The problem solves the following variables given a set of traffic demands $\mathcal{D}$:

• $\delta^r_{i,j} \in \{0, 1\}$: indicates whether the working path of demand $r$ traverses link $(i, j)$ (=1) or not (=0).

• $\eta^r_{i,j} \in \{0, 1\}$: indicates whether the protection path of demand $r$ traverses link $(i, j)$ (=1) or not (=0).

Objective

$$\text{minimize} \left\{ \sum_{r \in \mathcal{D}} \sum_{(i,j) \in \mathcal{L}} (\delta^r_{i,j} + \eta^r_{i,j}) \times n_r \right\} \quad (2.1)$$

Subject to: $(r \in \mathcal{D}, (i, j) \in \mathcal{L}, \lambda \in \mathcal{K}$, if not specified otherwise)

Flow conservation constraints on working paths:

$$\sum_{\forall o: (s_r, o) \in \mathcal{L}} \delta^r_{s_r,o} = 1, \delta^r_{i,s_r} = 0, \forall i : (i, s_r) \in \mathcal{L} \quad (2.2)$$

$$\sum_{\forall o: (d_r,o) \in \mathcal{L}} \delta^r_{d_r,o} = 0, \forall o : (d_r,o) \in \mathcal{L} \quad (2.3)$$

$$\sum_{\forall i: (i,j) \in \mathcal{L}} \delta^r_{i,j} - \sum_{\forall o: (j,o) \in \mathcal{L}} \delta^r_{j,o} = 0, \forall j \in \mathcal{N} (j \neq s_r, d_r) \quad (2.4)$$

Flow conservation constraints on protection paths:

$$\sum_{\forall o: (s_r, o) \in \mathcal{L}} \eta^r_{s_r,o} = 1, \eta^r_{i,s_r} = 0, \forall i : (i, s_r) \in \mathcal{L} \quad (2.5)$$

$$\sum_{\forall o: (d_r,o) \in \mathcal{L}} \eta^r_{d_r,o} = 0, \forall o : (d_r,o) \in \mathcal{L} \quad (2.6)$$

$$\sum_{\forall i: (i,j) \in \mathcal{L}} \eta^r_{i,j} - \sum_{\forall o: (j,o) \in \mathcal{L}} \eta^r_{j,o} = 0, \forall j \in \mathcal{N} (j \neq s_r, d_r) \quad (2.7)$$
The working path and protection path of demand \( r \) should be link-disjoint:

\[ \delta_{i,j}^r + \eta_{i,j}^r \leq 1 \]  \hspace{1cm} (2.8)

The number of wavelengths used on each link is subject to the physical constraint:

\[ \sum_{\forall r \in \mathcal{D}} (\delta_{i,j}^r + \eta_{i,j}^r) \times n_r \leq |\mathcal{K}|. \]  \hspace{1cm} (2.9)

**ILP2 – Joint RWA ILP for SP**

This ILP is developed for joint RWA optimization using shared path protection under the static traffic model (i.e., holding time unaware). In this optimization model, all the demands are assumed unchanged over time. Protection paths of demands can share the same wavelength-link as long as their corresponding working paths are link-disjoint. We call demands link-joint if their working paths traverse some common physical links.

(1) This problem solves the following variables in addition to those defined in ILP1:

- \( A_{i,j}^{r,\lambda} \in \{0,1\} \): indicates whether the working path of demand \( r \) traverses link \((i,j)\) using wavelength \( \lambda \) (=1) or not (=0).

- \( B_{i,j}^{r,\lambda} \in \{0,1\} \): indicates whether the protection path of demand \( r \) traverses link \((i,j)\) using wavelength \( \lambda \) (=1) or not (=0).

- \( S_{i,j}^{r_p, r_q} \in \{0,1\} \): indicates whether demands \( r_p \) and \( r_q \) are link-joint with respect to link \((i,j)\) (=1) or not (=0). Since \( S_{i,j}^{r_p, r_q} \) and \( S_{i,j}^{r_q, r_p} \) are the same, we only consider \( S_{i,j}^{r_p, r_q} \) \((p < q)\) to reduce the number of constraints in the ILP. Hereafter, we assume that \( r_p \) and \( r_q \) are ordered in demand set \( \mathcal{D} \) without loss of generality.

- \( S^{r_p, r_q} \in \{0,1\} \): indicates whether demands \( r_p \) and \( r_q \) are link-joint (=1) or not (=0) \((p < q)\).

- \( X_{i,j}^{\lambda} \in \{0,1\} \): indicates whether some working paths or protection paths traverse link \((i,j)\) using wavelength \( \lambda \) (=1) or not (=0).
Objective

\[
\text{minimize} \left\{ \sum_{\forall (i,j) \in L} \sum_{\forall \lambda \in K} X_{i,j}^\lambda \right\}
\] (2.10)

Subject to: \((r \in \mathcal{D}, (i, j) \in \mathcal{L}, \lambda \in \mathcal{K}, \text{if not specified otherwise})\)

Flow conservation constraints on working paths:

\[
\sum_{\forall o:(s_r,o) \in L} \delta_{s_r,o}^r = 1, \quad \delta_{s_r,s_r}^r = 0, \quad \forall i : (i, s_r) \in \mathcal{L}
\] (2.11)

\[
\sum_{\forall o:(d_r,o) \in L} \delta_{d_r,o}^r = 1, \quad \delta_{d_r,d_r}^r = 0, \quad \forall o : (d_r, o) \in \mathcal{L}
\] (2.12)

\[
\sum_{\forall i : (i,s) \in L} \delta_{i,s}^r - \sum_{\forall o : (j,o) \in L} \delta_{j,o}^r = 0, \quad \forall j \in \mathcal{N}(j \neq s_r,d_r)
\] (2.13)

Flow conservation constraints on protection paths:

\[
\sum_{\forall o:(s_r,o) \in L} \eta_{s_r,o}^r = 1, \quad \eta_{s_r,s_r}^r = 0, \quad \forall i : (i, s_r) \in \mathcal{L}
\] (2.14)

\[
\sum_{\forall o:(d_r,o) \in L} \eta_{d_r,o}^r = 1, \quad \eta_{d_r,d_r}^r = 0, \quad \forall o : (d_r, o) \in \mathcal{L}
\] (2.15)

\[
\sum_{\forall i : (i,j) \in L} \eta_{i,j}^r - \sum_{\forall o : (j,o) \in L} \eta_{j,o}^r = 0, \quad \forall j \in \mathcal{N}(j \neq s_r,d_r)
\] (2.16)

The working path and protection path of demand \(r\) should be link-disjoint:

\[
\delta_{i,j}^r + \eta_{i,j}^r \leq 1
\] (2.17)

Requested capacity of demand \(r\) should be satisfied:

\[
\sum_{\forall \lambda \in K} A_{i,j}^{r,\lambda} = \delta_{i,j}^r \times n_r, \quad \sum_{\forall \lambda \in K} B_{i,j}^{r,\lambda} = \eta_{i,j}^r \times n_r
\] (2.18)

Constraints indicating whether demands \(r_p\) and \(r_q\) are link-joint with respect to link \((i, j)\) \((S_{i,j}^{r_p,r_q}\) is set to 1 only when both \(\delta_{i,j}^{r_p}\) and \(\delta_{i,j}^{r_q}\) take on 1):\n
\[
\delta_{i,j}^{r_p} + \delta_{i,j}^{r_q} \leq S_{i,j}^{r_p,r_q} + 1, \quad \delta_{i,j}^{r_p} + \delta_{i,j}^{r_q} \geq 2 \times S_{i,j}^{r_p,r_q}, \quad \forall r_p, r_q \in \mathcal{D}(p < q)
\] (2.19)

Constraints indicating whether demands \(r_p\) and \(r_q\) are link-joint with respect to any link:

\[
S_{i,j}^{r_p,r_q} \leq \sum_{\forall (i,j) \in L} S_{i,j}^{r_p,r_q}, |L| \times S_{i,j}^{r_p,r_q} \geq \sum_{\forall (i,j) \in L} S_{i,j}^{r_p,r_q}, \forall r_p, r_q \in \mathcal{D}(p < q)
\] (2.20)
Wavelength $\lambda$ on link $(i,j)$ should not be shared if it has been used by a working path:

$$A_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda} \leq 1, \quad A_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda} \leq 1, \quad B_{i,j}^{r_p,\lambda} + A_{i,j}^{r_q,\lambda} \leq 1, \quad \forall r_p, r_q \in D(p < q) \quad (2.21)$$

Wavelength $\lambda$ on link $(i,j)$ should not be shared by two protection paths if their corresponding demands are link-joint:

$$B_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda} + S_{r_p, r_q} \leq 2, \quad \forall r_p, r_q \in D(p < q) \quad (2.22)$$

Constraints indicating whether wavelength $\lambda$ on link $(i,j)$ is used by some working paths or protection paths:

$$X_{i,j}^{\lambda} \leq \sum_{\forall r \in D} (A_{i,j}^{r,\lambda} + B_{i,j}^{r,\lambda}), \quad |D| \times X_{i,j}^{\lambda} \geq \sum_{\forall r \in D} (A_{i,j}^{r,\lambda} + B_{i,j}^{r,\lambda}). \quad (2.23)$$

**ILP3 – Joint RWA ILP for DP-S**

This ILP is developed for joint RWA optimization using dedicated path protection under the *scheduled traffic model* (i.e., holding time aware). In this optimization model, demands only last during specified time intervals (i.e., holding time aware) and hence network resources can be reused among demands which are time-disjoint. However, all protection paths of the demands that overlap in time are not allowed to share resources even their corresponding working paths are link-disjoint, therefore termed as dedicated protection.

(1) This problem uses the following inputs in addition to those defined in **ILP1**:

- $D$: the set of scheduled traffic demands; For each demand $r = (s_r, d_r, n_r, \alpha_r, \beta_r) \in D$, $s_r$, $d_r$, $n_r$, $\alpha_r$, and $\beta_r$ are the source node, destination node, the number of requested lightpaths, starting and ending time of demand $r$, respectively.

- $T_{r_p, r_q} \in \{0, 1\}$: indicates whether demands $r_p$ and $r_q$ overlap in time ($=1$) or not ($=0$) $(p < q)$. 

26
Objective

\[
\text{minimize} \left\{ \sum_{\forall (i,j) \in L \forall \lambda \in K} X_{i,j}^{\lambda} \right\} \tag{2.24}
\]

\textbf{Subject to:} \ (r \in \mathcal{D}, (i, j) \in \mathcal{L}, \lambda \in \mathcal{K}, \text{ if not specified otherwise} )

Flow conservation constraints on working paths:

\[
\sum_{\forall o: (s_r, o) \in L} \delta_{s_r,o}^r = 1, \ \delta_{i,s}^r = 0, \ \forall i : (i, s_r) \in \mathcal{L} \tag{2.25}
\]

\[
\sum_{\forall i: (i,d_r) \in L} \delta_{i,d_r}^r = 1, \ \delta_{d_r,o}^r = 0, \ \forall o : (d_r, o) \in \mathcal{L} \tag{2.26}
\]

\[
\sum_{\forall i: (i,j) \in L} \delta_{i,j}^r - \sum_{\forall o: (j,o) \in L} \delta_{j,o}^r = 0, \ \forall j \in \mathcal{N}(j \neq s_r, d_r) \tag{2.27}
\]

Flow conservation constraints on protection paths:

\[
\sum_{\forall o: (s_r, o) \in L} \eta_{s_r,o}^r = 1, \ \eta_{i,s_r}^r = 0, \ \forall i : (i, s_r) \in \mathcal{L} \tag{2.28}
\]

\[
\sum_{\forall i: (i,d_r) \in L} \eta_{i,d_r}^r = 1, \ \eta_{d_r,o}^r = 0, \ \forall o : (d_r, o) \in \mathcal{L} \tag{2.29}
\]

\[
\sum_{\forall i: (i,j) \in L} \eta_{i,j}^r - \sum_{\forall o: (j,o) \in L} \eta_{j,o}^r = 0, \ \forall j \in \mathcal{N}(j \neq s_r, d_r) \tag{2.30}
\]

The working path and protection path of demand \( r \) should be link-disjoint:

\[
\delta_{i,j}^r + \eta_{i,j}^r \leq 1 \tag{2.31}
\]

Requested capacity of demand \( r \) should be satisfied:

\[
\sum_{\forall \lambda \in \mathcal{K}} A_{i,j}^{r,\lambda} = \delta_{i,j}^r \times n_r, \ \sum_{\forall \lambda \in \mathcal{K}} B_{i,j}^{r,\lambda} = \eta_{i,j}^r \times n_r \tag{2.32}
\]

Wavelength \( \lambda \) on link \((i, j)\) should not be shared by two demands if they overlap in time:

\[
A_{i,j}^{r_p,\lambda} + A_{i,j}^{r_q,\lambda} \leq 1, \ A_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda} \leq 1, \ B_{i,j}^{r_p,\lambda} + A_{i,j}^{r_q,\lambda} \leq 1, \ B_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda} \leq 1,
\]

\[
\forall r_p, r_q \in \mathcal{D}(p < q \text{ and } T_{r_p,r_q} = 1) \tag{2.33}
\]
Constraints indicating whether wavelength $\lambda$ on link $(i,j)$ is used by some working paths or protection paths:

$$X_{i,j}^\lambda \leq \sum_{r \in D} (A_{i,j}^r + B_{i,j}^r), \quad |D| \times X_{i,j}^\lambda \geq \sum_{r \in D} (A_{i,j}^r + B_{i,j}^r).$$

(2.34)

Note that in this formulation, wavelength-links of the working path of demand $r$ can be reused by the working path or protection path of other demands that do not overlap in time with $r$. The same is also true for wavelength-links of the protection path of a demand.

**ILP4 – Joint RWA ILP for SP-S**

This ILP is developed for joint RWA optimization using shared path protection under the scheduled traffic model (i.e., holding time aware). This optimization model enables resource optimization in both space and time, i.e, network resources can be reused among demands that are time-disjoint and can also be shared by protection paths whose corresponding working paths are link-disjoint, even their demands overlap in time.

(1) This problem uses the following inputs in addition to those defined in **ILP2**:

- $D$: the set of scheduled traffic demands; For each demand $r = (s_r, d_r, n_r, \alpha_r, \beta_r) \in D$, $s_r$, $d_r$, $n_r$, $\alpha_r$, and $\beta_r$ are the source node, destination node, the number of requested lightpaths, starting and ending time of demand $r$, respectively.

(2) This problem solves the following variables in addition to those defined in **ILP2**:

- $TS^{r_p-r_q} \in \{0, 1\}$: indicates whether demands $r_p$ and $r_q$ overlap in time and link-joint (=1) or not (=0) ($p < q$).

**Objective**

$$\text{minimize} \sum_{\forall (i,j) \in L} \sum_{\forall \lambda \in K} X_{i,j}^\lambda$$

(2.35)

**Subject to:** ($r \in D, (i,j) \in L, \lambda \in K$, if not specified otherwise)
Flow conservation constraints on working paths:

\[
\sum_{\forall o:(s_r,o)\in L} \delta^r_{s_r,o} = 1, \delta^r_{i,s_r} = 0, \forall i : (i, s_r) \in L \tag{2.36}
\]

\[
\sum_{\forall i:(i,d_r)\in L} \delta^r_{i,d_r} = 1, \delta^r_{d_r,o} = 0, \forall o : (d_r,o) \in L \tag{2.37}
\]

\[
\sum_{\forall i:(i,j)\in L} \delta^r_{i,j} - \sum_{\forall o:(j,o)\in L} \delta^r_{j,o} = 0, \forall j \in N(j \neq s_r, d_r) \tag{2.38}
\]

Flow conservation constraints on protection paths:

\[
\sum_{\forall o:(s_r,o)\in L} \eta^r_{s_r,o} = 1, \eta^r_{i,s_r} = 0, \forall i : (i, s_r) \in L \tag{2.39}
\]

\[
\sum_{\forall i:(i,d_r)\in L} \eta^r_{i,d_r} = 1, \eta^r_{d_r,o} = 0, \forall o : (d_r,o) \in L \tag{2.40}
\]

\[
\sum_{\forall i:(i,j)\in L} \eta^r_{i,j} - \sum_{\forall o:(j,o)\in L} \eta^r_{j,o} = 0, \forall j \in N(j \neq s_r, d_r) \tag{2.41}
\]

The working path and protection path of demand \(r\) should be link-disjoint:

\[
\delta^r_{i,j} + \eta^r_{i,j} \leq 1 \tag{2.42}
\]

Requested capacity of demand \(r\) should be satisfied:

\[
\sum_{\forall \lambda \in K} A^r_{i,j} = \delta^r_{i,j} \times n_r, \sum_{\forall \lambda \in K} B^r_{i,j} = \eta^r_{i,j} \times n_r \tag{2.43}
\]

Constraints indicating whether demands \(r_p\) and \(r_q\) are link-joint with respect to link \((i,j)\) (\(S^r_{i,j}\) is set to 1 only when both \(\delta^r_{i,j}\) and \(\delta^r_{i,j}\) take on 1):

\[
\delta^r_{i,j} + \delta^q_{i,j} \leq S^r_{i,j} + 1, \delta^r_{i,j} + \delta^q_{i,j} \geq 2 \times S^r_{i,j}, \forall r_p, r_q \in D(p < q) \tag{2.44}
\]

Constraints indicating whether demands \(r_p\) and \(r_q\) overlap in time and link-joint:

\[
TS^r_{i,j} \leq T^r_{i,j} \times \sum_{\forall (i,j) \in L} S^r_{i,j} \times |L| \times TS^r_{i,j} \geq T^r_{i,j} \times \sum_{\forall (i,j) \in L} S^r_{i,j}, \forall r_p, r_q \in D(p < q) \tag{2.45}
\]
Wavelength $\lambda$ on link $(i, j)$ should not be used by two demands if they overlap in time:

$$A_{r_p}^{\lambda} + A_{r_q}^{\lambda} \leq 1, \ A_{i,j}^{\lambda} + B_{i,j}^{\lambda} \leq 1, \ B_{r_p}^{\lambda} + A_{r_q}^{\lambda} \leq 1, \ \forall r_p, r_q \in D(p < q \text{ and } T_{r_p-r_q} = 1)$$

(2.46)

Wavelength $\lambda$ on link $(i, j)$ should not be shared by two protection paths if their corresponding demands overlap in time and link-joint:

$$B_{r_p}^{\lambda} + B_{r_q}^{\lambda} + TS_{r_p-r_q} \leq 2, \ \forall r_p, r_q \in D(p < q)$$

(2.47)

Constraints indicating whether wavelength $\lambda$ on link $(i, j)$ is used by some working paths or protection paths:

$$X_{i,j}^{\lambda} \leq \sum_{r \in D} (A_{r,i,j}^{\lambda} + B_{r,i,j}^{\lambda}), \ |D| \times X_{i,j}^{\lambda} \geq \sum_{r \in D} (A_{r,i,j}^{\lambda} + B_{r,i,j}^{\lambda})$$

(2.48)

Note that this formulation maximally exploits the network resource reuse in both space and time, resulting in the minimization of total network resources used.

### 2.2.2 Two-step Optimization Approach

In this subsection, we propose a two-step optimization approach for protection architectures SP, DP-S and SP-S. We separate the joint routing and wavelength assignment problem into a routing subproblem and a wavelength assignment subproblem. We use an ILP (ILP5) to compute a pair of link-disjoint routes for each of the demands in $D$ as its working path and protection path. The objective is to minimize the total cost of a pair of routes (i.e., total hop count over the pair of routes). The resulting routing information is recorded into variables $\delta_{r,i,j}$ and $\eta_{r,i,j}$. For the wavelength-assignment subproblem, $\delta_{r,i,j}$ and $\eta_{r,i,j}$ will be used as input. In addition, since the working path and protection path of each demand in $D$ have been determined in the routing subproblem, we can compute $S_{r_p-r_q}$ and use it as input for wavelength assignment optimization (e.g., in ILP6 that optimizes wavelength...
assignment for SP). Notice that in ILP2, ILP6's counterpart in the joint RWA optimization approach, $S^{r_p, r_q}$ has to be determined in the ILP formulations. Similarly, $T^{r_p, r_q}$ can also be computed in advance based on the knowledge of $T^{r_p, r_q}$ and $S^{r_p, r_q}$, and will be used as input in ILP8 that optimizes wavelength assignment for SP-S.

The objectives of the wavelength assignment subproblems are to obtain proper wavelength assignments such that the total network resources used are minimized. The definitions of many variables used in the following formulations are the same as those in Section 2.2.1.

**ILP5 — ILP for the Routing Subproblem**

This ILP is developed to pre-compute a pair of link-disjoint routes for each of the demands in $D$ as its working path and protection path. In the objective function, we use a coefficient $M$ for $\delta_{i,j}^r$ to make the selection of the pair of routes more flexible. When $M$ takes on 1, for instance, the shortest pair of routes will be chosen for each demand. In the simulation, we use a large coefficient for $M$ to enforce that the shorter of the pair of routes is selected as the working path.

**Objective**

$$\text{minimize}\{ \sum_{r \in D} \sum_{(i,j) \in L} (M \times \delta_{i,j}^r + \eta_{i,j}^r) \} \quad (2.49)$$

**Subject to:** $(r \in D, (i, j) \in L, \text{if not specified otherwise})$

Flow conservation constraints on working paths:

$$\sum_{\forall o : (s_r, o) \in L} \delta_{s_r,o}^r = 1, \quad \delta_{i,s_r}^r = 0, \quad \forall i : (i, s_r) \in L \quad (2.50)$$

$$\sum_{\forall o : (d_r, o) \in L} \delta_{i,d_r}^r = 1, \quad \delta_{d_r,o}^r = 0, \quad \forall o : (d_r, o) \in L \quad (2.51)$$

$$\sum_{\forall o : (j,o) \in L} \delta_{i,j}^r - \sum_{\forall o : (j,o) \in L} \delta_{j,o}^r = 0, \quad \forall j \in N(j \neq s_r, d_r) \quad (2.52)$$

Flow conservation constraints on protection paths:

$$\sum_{\forall o : (s_r, o) \in L} \eta_{s_r,o}^r = 1, \quad \eta_{i,s_r}^r = 0, \quad \forall i : (i, s_r) \in L \quad (2.53)$$
\[
\sum_{\forall i : (i, d_r) \in L} \eta_{i, d_r}^r = 1, \eta_{d_r, o}^r = 0, \forall o : (d_r, o) \in L \tag{2.54}
\]

\[
\sum_{\forall i : (i, j) \in L} \eta_{i, j}^r - \sum_{\forall o : (j, o) \in L} \eta_{j, o}^r = 0, \forall j \in N(j \neq s_r, d_r) \tag{2.55}
\]

The working path and protection path of demand \( r \) should be link-disjoint:

\[
\delta_{i, j}^r + \eta_{i, j}^r \leq 1 \tag{2.56}
\]

**ILP6 — Wavelength Assignment ILP for SP**

This ILP is developed for wavelength assignment using shared path protection under the static traffic model (i.e., holding time unaware).

**Objective**

\[
\text{minimize} \left\{ \sum_{\forall (i, j) \in L} \sum_{\forall \lambda \in K} X_{i, j}^\lambda \right\} \tag{2.57}
\]

**Subject to:** \(( r \in D, (i, j) \in L, \lambda \in K, \text{if not specified otherwise})\)

Requested capacity of demand \( r \) should be satisfied:

\[
\sum_{\forall \lambda \in K} A_{i, j}^r, \lambda = \delta_{i, j}^r \times n_r, \sum_{\forall \lambda \in K} B_{i, j}^r, \lambda = \eta_{i, j}^r \times n_r \tag{2.58}
\]

Wavelength \( \lambda \) on link \((i, j)\) should not be shared if it has been used by a working path:

\[
A_{i, j}^{r_p, \lambda} + A_{i, j}^{r_q, \lambda} \leq 1, A_{i, j}^{r_p, \lambda} + B_{i, j}^{r_q, \lambda} \leq 1, B_{i, j}^{r_p, \lambda} + A_{i, j}^{r_q, \lambda} \leq 1, \forall r_p, r_q \in D(p < q) \tag{2.59}
\]

Wavelength \( \lambda \) on link \((i, j)\) should not be shared by two protection paths if their corresponding demands are link-joint:

\[
B_{i, j}^{r_p, \lambda} + B_{i, j}^{r_q, \lambda} \leq 1, \forall r_p, r_q \in D(p < q \text{ and } S_{r_p, r_q} = 1) \tag{2.60}
\]

Constraints indicating whether wavelength \( \lambda \) on link \((i, j)\) is used by some working paths or protection paths:

\[
X_{i, j}^\lambda \leq \sum_{\forall r \in D} (A_{i, j}^{r, \lambda} + B_{i, j}^{r, \lambda}), |D| \times X_{i, j}^\lambda \geq \sum_{\forall r \in D} (A_{i, j}^{r, \lambda} + B_{i, j}^{r, \lambda}). \tag{2.61}
\]

32
ILP7 – Wavelength Assignment ILP for DP-S

This ILP is developed for wavelength assignment using dedicated path protection under the scheduled traffic model (i.e., demands ∈ \(\mathcal{D}\) are holding time aware).

Objective

\[
\text{minimize}\left\{ \sum_{(i,j) \in \mathcal{L}} \sum_{\lambda \in \mathcal{K}} X^\lambda_{i,j} \right\}
\]

Subject to: \((r \in \mathcal{D}, (i, j) \in \mathcal{L}, \lambda \in \mathcal{K}, \text{if not specified otherwise})\)

Requested capacity of demand \(r\) should be satisfied:

\[
\sum_{\lambda \in \mathcal{K}} A^r_{i,j} = \delta^r_{i,j} \times n_r, \quad \sum_{\lambda \in \mathcal{K}} B^r_{i,j} = \eta^r_{i,j} \times n_r
\]

(2.63)

Wavelength \(\lambda\) on link \((i, j)\) should not be shared by two demands if they overlap in time:

\[
A^r_{i,j} + A^{r'}_{i,j} \leq 1, \quad A^r_{i,j} + B^{r'}_{i,j} \leq 1, \quad B^r_{i,j} + A^{r'}_{i,j} \leq 1, \quad B^r_{i,j} + B^{r'}_{i,j} \leq 1,
\]

\[\forall r_p, r_q \in \mathcal{D}(p < q \text{ and } T^{r_p \rightarrow r_q} = 1)\]

(2.64)

Constraints indicating whether wavelength \(\lambda\) on link \((i, j)\) is used by some working paths or protection paths:

\[
X^\lambda_{i,j} \leq \sum_{r \in \mathcal{D}} (A^r_{i,j} + B^r_{i,j}), \quad |\mathcal{D}| \times X^\lambda_{i,j} \geq \sum_{r \in \mathcal{D}} (A^r_{i,j} + B^r_{i,j}).
\]

(2.65)

ILP8 – Wavelength Assignment ILP for SP-S

This ILP is developed for wavelength assignment using shared path protection under the scheduled traffic model (i.e., demands ∈ \(\mathcal{D}\) are holding time aware).

Objective

\[
\text{minimize}\left\{ \sum_{(i,j) \in \mathcal{L}} \sum_{\lambda \in \mathcal{K}} X^\lambda_{i,j} \right\}
\]

Subject to: \((r \in \mathcal{D}, (i, j) \in \mathcal{L}, \lambda \in \mathcal{K}, \text{if not specified otherwise})\)

Requested capacity of demand \(r\) should be satisfied:

\[
\sum_{\lambda \in \mathcal{K}} A^r_{i,j} = \delta^r_{i,j} \times n_r, \quad \sum_{\lambda \in \mathcal{K}} B^r_{i,j} = \eta^r_{i,j} \times n_r
\]

(2.67)

33
Wavelength $\lambda$ on link $(i, j)$ should not be used by two demands if they overlap in time:

$$A_{i,j}^{p,\lambda} + A_{i,j}^{q,\lambda} \leq 1, \ A_{i,j}^{p,\lambda} + B_{i,j}^{r,\lambda} \leq 1, \ B_{i,j}^{p,\lambda} + A_{i,j}^{r,\lambda} \leq 1, \ \forall r_p, r_q \in \mathcal{D}(p < q \text{ and } T_{r_p, r_q} = 1)$$ (2.68)

Wavelength $\lambda$ on link $(i, j)$ should not be shared by two protection paths if their corresponding demands overlap in time and link-joint:

$$B_{i,j}^{p,\lambda} + B_{i,j}^{q,\lambda} \leq 1, \ \forall r_p, r_q \in \mathcal{D}(p < q \text{ and } TS_{r_p, r_q} = 1)$$ (2.69)

Constraints indicating whether wavelength $\lambda$ on link $(i, j)$ is used by some working paths or protection paths:

$$X_{i,j}^{\lambda} \leq \sum_{\forall r \in \mathcal{D}} (A_{r_{ij}}^{p,\lambda} + B_{r_{ij}}^{r,\lambda}), \ |\mathcal{D}| \times X_{i,j}^{\lambda} \geq \sum_{\forall r \in \mathcal{D}} (A_{r_{ij}}^{p,\lambda} + B_{r_{ij}}^{r,\lambda}).$$ (2.70)

2.2.3 Numerical and Simulation Results

We present and evaluate the performance of the ILP formulations described in this section. The objective of all the optimization models is to determine a pair of working path and protection path, and assign wavelengths to them for each traffic demand in a demand set, so that the total number of wavelength-links used is minimized given that network resources are sufficient to accommodate all demands.

We first evaluate the joint RWA ILPs, ILP1 through ILP4, for all the four protection architectures (i.e., DP, SP, DP-S and SP-S), and the two-step optimization approach for SP, DP-S and SP-S (i.e., ILP6, ILP7, and ILP8 after ILP5 is performed). These different optimization models are denoted by ILP1 through ILP4 and ILP6 through ILP8, respectively, in Table 2.2.

We use the three networks used in [1] (Fig. 5 (a), (b) and (c) of [1]), which are re-drawn in Fig. 2.1, and the NSFNET topology shown in Fig. 2.2 for performance evaluation and comparison. We assume that links in the example networks are directed by replacing a link with a pair of directed links. The source and destination
of a demand are generated randomly, and the bandwidth requirements in terms of number of lightpaths is drawn from a uniform distribution in [1,3]. In addition, the setup and teardown times of a demand are also generated randomly between 0 and 24 hours, and meet the demand time correlation requirements. We use the demand time correlation defined in [50] to characterize the time overlapping behavior among a set of demands. We consider three classes of demand set with a demand time correlation factor being weak (0.01), medium (0.1), and strong (0.5) to measure the extent of time overlapping among demands in a set. The ILP optimization problems are solved using CPLEX 8.1 running on a 2.5 GHz Pentium IV processor with 2 GB RAM. Feasible sub-optimal solutions are recorded after 4 hours of execution if optimal solutions are not obtained before the time limit. In Table 2.2, numbers with asterisks indicate the optimal solution found and numbers without asterisks indicate the current best solution reported by CPLEX within the time limit.

![Networks](image)

(a) A 3-node network  (b) A 6-node network  (c) A 10-node network

*Fig. 2.1: Three example networks used in [1].*

![NSFNET Topology](image)

*Fig. 2.2: The NSFNET topology.*

We investigate six scenarios in the simulations. The basic information associated
with each scenario such as the network topology used, the number of demands \((D)\), and the number of wavelengths on each link in the network \((K)\) is listed in Table 2.1. Table 2.2 shows the total number of wavelength-links needed to satisfy different traffic demand sets in different optimization models with weak, medium, and strong demand time correlation, respectively. From the table, we observe that in the first two cases, the size of example networks is small; the set of traffic demands is small; and the number of wavelengths on each link is not large. Under such scenarios, all the optimization models investigated are able to achieve the optimal solutions. \textbf{ILP1}, \textbf{ILP6}, \textbf{ILP7} and \textbf{ILP8} can also achieve optimal solutions under Scenario 3. In other cases with larger networks and large demand sets, however, all optimization models except \textbf{ILP1} cannot achieve the optimal solutions, even though some of them (\textbf{ILP6}, \textbf{ILP7} and \textbf{ILP8}) employ the two-step optimization approach, as shown in the last three cases of Table 2.2.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Network</th>
<th>(D)</th>
<th>(K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3-node</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>6-node</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>10-node</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>10-node</td>
<td>16</td>
<td>32</td>
</tr>
<tr>
<td>5</td>
<td>14-node</td>
<td>16</td>
<td>32</td>
</tr>
<tr>
<td>6</td>
<td>14-node</td>
<td>32</td>
<td>64</td>
</tr>
</tbody>
</table>

\textit{Tab. 2.1: Scenario Information.}

From the table, we observe that \textbf{DP-S} and \textbf{SP-S} schemes use much fewer wavelength-links than \textbf{DP} and \textbf{SP} schemes in all scenarios, especially when the demand time correlation is weak. For example, in Case 3, the performance improvement of \textbf{ILP7} over \textbf{ILP1} is 49\%, 39\% and 25\% when the demand time correlations are weak, medium and strong, respectively, while that of \textbf{ILP8} over \textbf{ILP6} is 45\%, 34\% and 18\%, respectively. This is because the protection schemes under the scheduled traffic model are able to exploit the time-disjointness among demands and reuse wavelength-links as much as possible.
From the percentages of performance improvement shown above, we observe that the improvement of DP-S over DP is more significant than that of SP-S over SP. Moreover, the improvement of SP over DP under the static traffic model (i.e., holding time unaware) appears to be larger than that of SP-S over DP-S under the scheduled traffic model. For example, the improvement of ILP2 over ILP1 and that of ILP4 over ILP3 are 4% and 0%, respectively, in Case 5 with weak demand time correlation. The reason may be that, in the conventional static traffic model, the resource utilization can be improved significantly through backup resource sharing, however the likelihood of finding sharable wavelength-links for time-disjoint protection paths is smaller in SP-S scheme. Therefore, we observe less significant improvement of SP-S over DP-S, and a dedicated protection scheme can achieve greater improvement than a shared protection scheme under the scheduled traffic model. However, as the demand time correlation gets stronger, the improvement of SP-S over DP-S increases as well. For instance, the improvement of ILP8 over ILP7 is 1%, 13% and
18% in Case 6 when demand time correlations are weak, medium and strong, respectively. This is because when the demand time correlation gets stronger, it becomes more difficult for \textbf{DP-S} to reuse network resources among demands, and thus the advantage of \textbf{SP-S} amplifies since it is able to maximally exploit network resource reuse in both the space and time domains.

\section{2.3 Heuristic Algorithm}

In this section, we propose a capacity provision matrix based Iterative Survivable Routing (ISR) algorithm to solve the survivable service provisioning problem under the scheduled traffic model by processing demands sequentially. The algorithm runs iteratively to approximate the optimal solution. A similar matrix based method was used in \cite{65} where the conventional static traffic model was considered; and the objective was to minimize only the protection capacity used. The problem considered in this work is different. Our proposed algorithm strives to minimize the total network resources used by \textit{both working paths and protection paths}\footnote{That is, both working capacity and protection capacity are minimized simultaneously.} of a given set of demands through exploiting network resource reuse in both space and time domains simultaneously.

We first describe the capacity provision matrix based optimization model used in our approach, and then explain how to determine the capacity provision matrix which is the core step of our ISR algorithm. The ISR algorithm processes demands sequentially; and the order of processing is determined by various demand ordering policies detailed in Section 2.3.3.

\subsection{2.3.1 Capacity Provision Matrix based Optimization Model}

Note that the objective of our approach is to minimize the total network resources (i.e., number of wavelength-links) used by working paths and protection paths of
Fig. 2.3: Illustration of capacity provision matrix based optimization model. The numbers besides the links are used to number the links.

a given set of demands while 100% restorability is guaranteed against any single failure. Demands are processed sequentially. Given a demand, our approach tries to accommodate the demand by finding paths that will use the least amount of additional network resource after using sharable resources as much as possible. Once a working path and a protection path are found for each demand in the given demand set, the total network resources used can be determined. The algorithm then runs iteratively to reduce the total resources used.

Notation used is defined as follows (see Fig. 2.3 and Example 2.3.1 for a better explanation). Two $1 \times L$ binary row vectors $a_r = \{a_{rl}\}$ and $b_r = \{b_{rl}\}$ are used to represent the working path and the protection paths of demand $r$, respectively. The $l$-th element in vector $a_r$ ($b_r$) of demand $r$ takes on 1 if and only if the working (protection) path passes through link $l$. Stacking of these row vectors forms two $D \times L$ matrices $A = \{a_r\}$ and $B = \{b_r\}$ which represent the working path link incidence matrix and protection path link incidence matrix of the given demand set, respectively.

Let $G = \{g_{lk}\}_{L \times L}$ denote the capacity provision matrix whose elements $g_{lk}$ indicate
the minimum total capacity required on link $l$ when link $k$ fails. Note that the total capacity, $g_{lk}$, includes the capacity required by all the working paths passing through link $l$ and the capacity on link $l$ required by the protection paths to protect their corresponding working paths that use link $k$. Determining the capacity provision matrix is the most critical step of our ISR algorithm. We will explain the calculation of $G$ in details in Section 2.3.2 and prove that the capacity determined by our approach is the minimum total capacity required on link $l$ when link $k$ fails.

After obtaining matrix $G$, we use a column vector $s = \{s_l\}_{L \times 1}$ to denote the minimum total capacity required on each link. $s$ is obtained using Eqn. (2.71). The function max in Eqn. (2.71) takes the maximum element in each row of $G$ as the corresponding element in $s$, which is represented in Eqn. (2.72). As will be shown in Section 2.3.2, the minimum total capacity required on link $l$ ($s_l$) is always sufficient and also necessary to accommodate all working paths on link $l$ and to protect any single link failure in the entire network.

\[
s = \max G, \tag{2.71}
\]

\[
s_l = \max_{k=1}^L g_{lk}, \ 1 \leq l \leq L. \tag{2.72}
\]

Let $\Lambda$ denote the total number of wavelength-links used by working paths and protection paths of all demands; and $e$ be a unit column vector of size $L$. Then $\Lambda$ can be calculated as $e^T s$ in Eqn. (2.73). Given a set of scheduled demand $D$ and the network $G$, the objective of our proposed approach is therefore to minimize $\Lambda$:

\[
\Lambda = e^T s. \tag{2.73}
\]

We use an example to explain the notation further. Two static demands are routed in the network of Fig. 2.3. The figure illustrates the capacity provision matrix $G$, the vector of minimum required capacity of links $s$, and the total number of
wavelength-links Λ used to accommodate the two demands. The working path and protection path of demand 1 between nodes 0 and 1 are WP1 and PP1. The working path and protection path of demand 2 between nodes 0 and 2 are WP2 and PP2. Only six links in the network are used by the paths of the two demands. Therefore, only these links are included in the matrices to reduce the size of the matrices in the example. The numbers besides the links are used to number the links. Path link incidence matrices A and B give the working paths and protection paths used by the two demands. The capacity provision matrix G can be easily calculated by observation in this simple example. The calculation of s and Λ is also straightforward.

2.3.2 Determination of Capacity Provision Matrix

Determining the capacity provision matrix G is the most critical step because it has direct impact on the values of s and Λ. In this subsection, we explain how to calculate \( G = \{g_{lk}\}_{L \times L} \); and then prove that each element of G, \( g_{lk} \), is the minimum total capacity required on link l to accommodate all working paths on link l and to protect the failure of link k.

Let \( \mu_l \) and \( \nu_{lk} \) be the minimum capacity required by working paths passing through link l and the minimum capacity on link l required by protection paths when link k fails, respectively. Under the conventional static traffic model in which the connection-holding-time is infinite, all demands can be assumed to have the same setup time and same teardown time. Obviously, \( g_{lk} = \mu_l + \nu_{lk} \) in such a model. Under the scheduled traffic model, however, some working paths and protection paths using link l may be disjoint in time; and the property of time-disjointness can then be exploited to reduce the total capacity required on link l by sharing the network resource between working paths and protection paths. This leads to \( g_{lk} \leq \mu_l + \nu_{lk} \).

Next, we will first explain how to determine the minimum capacity needed to ac-
commodate a set of scheduled demands by maximally exploiting the time disjointness among the demands; and then demonstrate how to calculate $\mu_l$, $\nu_{lk}$ and $g_{lk}$.

Under the scheduled traffic model, since the setup time and teardown time of each demand are known in advance, we are able to determine the time correlation of any two demands (overlapping or disjointness) before their services are provisioned. Given a given set of demands, $D$, an interval graph can be constructed in which each vertex corresponds to a demand in $D$; and two vertices are adjacent with an edge connecting them if and only if their corresponding demands overlap in time. In addition, each vertex in the graph is associated with a weight which is equal to the capacity requirement of its corresponding demand. We call such a graph a weighted interval graph. For example, let $D$ be the example demand set shown in Table 1.1. Fig. 2.4 shows the weighted interval graph corresponding to $D$. From the graph, we observe that the minimum total capacity needed on a link to accommodate all the demands using the link is equal to the total weight of a maximum weight clique of the graph. A clique is a set of vertices in the graph and the vertices in the set are pairwise adjacent. A clique is called a maximum weight clique if its weight, i.e., the sum of the weights of its vertices, is the largest among all the cliques in the graph. We use $\Omega(D)$ and $\omega(D)$ to denote the set of demands in the maximum weight clique and its total weight, respectively. From Fig. 2.4, we can easily determine that $\Omega(D) = \{r_2, r_3, r_4, r_5\}$ and $\omega(D) = 8$.

By exploiting the time disjointness among a given set of demands, we transform the problem of finding the minimum total capacity to accommodate all the demands to the problem of finding a maximum weight clique of the weighted interval graph corresponding to the demand set. The maximum weight clique problem is known to be $NP$-hard in an arbitrary graph [66]; however, it can be solved in time $O(n \log n)$ in an interval graph where $n$ is the number of vertices (i.e., the number of demands) [67].
To determine the capacity provision matrix $G$, we need to calculate its elements $g_{lk}$. Although the values of $\mu_l$ and $\nu_{lk}$ are not explicitly used in our ISR algorithm, we demonstrate how to calculate them in addition to $g_{lk}$ below. We investigate the demands whose working paths or protection paths pass through link $l$ and the demands whose working paths pass through link $k$. These demands can be divided into three groups as follows (shown in Fig. 2.5 with a simplified network):

- **Demand Group 0**: $\mathcal{DG}^0 = \{r^0_1, r^0_2, \ldots, r^0_{g_0}\}$ where $r^0_i (1 \leq i \leq g_0)$ is a demand whose working path goes through link $l$.

- **Demand Group 1**: $\mathcal{DG}^1 = \{r^1_1, r^1_2, \ldots, r^1_{g_1}\}$ where $r^1_i (1 \leq i \leq g_1)$ is a demand whose protection path goes through link $l$.

- **Demand Group 2**: $\mathcal{DG}^2 = \{r^2_1, r^2_2, \ldots, r^2_{g_2}\}$ where $r^2_i (1 \leq i \leq g_2)$ is a demand whose working path goes through link $k$.

Based on this demand classification, we prove a few theorems on the calculation
Theorem 1: Given a set of scheduled demands $\mathcal{D}$ that share link $l$, the minimum capacity required on link $l$ to accommodate all the demands is equal to the weight of the maximum weight clique of the weighted interval graph corresponding to $\mathcal{D}$, i.e., $\omega(\mathcal{D})$.

**Proof:** We prove the theorem by contradiction. Suppose the minimum capacity required to accommodate all the demands in $\mathcal{D}$ is $\delta$ and $\delta < \omega(\mathcal{D})$. Let $\mathcal{G}_l$ be the weighted interval graph corresponding to $\mathcal{D}$. Since $\Omega(\mathcal{D})$ is the set of demands corresponding to a maximum weight clique of $\mathcal{G}_l$, any two demands in $\Omega(\mathcal{D})$ must be overlapping in time. In other words, there must exist a time interval $[t_1, t_2]$ such that all the demands in $\Omega(\mathcal{D})$ fall in it. Consequently, at least $\omega(\mathcal{D})$ units of capacity is required to accommodate all demands in $\mathcal{D}$ during time interval $[t_1, t_2]$. This means $\delta \geq \omega(\mathcal{D})$ since we suppose $\delta$ is the minimum capacity required to accommodate all the demands in $\mathcal{D}$. This results in a contradiction.

Theorem 2: Given a set of scheduled demands $\mathcal{D}$, and network $\mathcal{G}$, the minimum capacity required to accommodate all the working paths using link $l$ is equal to the weight of the maximum weight clique of the weighted interval graph corresponding to Demand Group 0. That is, $\mu_l = \omega(\mathcal{DG}^0)$.

**Proof:** It is trivial that $\mu_l$ only depends on the set of demands whose working paths pass through link $l$, i.e., $\mathcal{DG}^0$. According to Theorem 1, we have $\mu_l = \omega(\mathcal{DG}^0)$. }

Theorem 3: The minimum capacity on link $l$ required by protection paths when link $k$ fails is equal to the weight of the maximum weight clique of the weighted interval graph corresponding to the demand set which includes the demands in both Demand Group 1 and Demand Group 2. That is, $\nu_{lk} = \omega(\mathcal{DG}^1 \cap \mathcal{DG}^2)$. 

44
Proof: We use \( \mathcal{DG}^i \setminus \mathcal{DG}^j \) to denote the demands in group \( \mathcal{DG}^i \) but not in \( \mathcal{DG}^j \). When link \( k \) fails, no capacity is required on link \( l \) to protect the working traffic of demands in \( \mathcal{DG}^2 \setminus \mathcal{DG}^1 \). On the other hand, there is no need to assign capacity to protection paths of demands in \( \mathcal{DG}^1 \setminus \mathcal{DG}^2 \) on link \( l \) to protect the failure of link \( k \) because their corresponding working paths do not pass through link \( k \). Only demands in \( \mathcal{DG}^1 \cap \mathcal{DG}^2 \) will affect the value of \( \nu_{lk} \). According to Theorem 1, we have \( \nu_{lk} = \omega(\mathcal{DG}^1 \cap \mathcal{DG}^2) \).

Theorem 4: The minimum total capacity required on link \( l \) to accommodate all working paths on link \( l \) and to protect the failure of link \( k \) is equal to the weight of the maximum weight clique of the weighted interval graph corresponding to the demand set which includes the demands in Demand Group 0 and those in both Demand Group 1 and Demand Group 2. That is, \( g_{lk} = \omega(\mathcal{DG}^0 \cup (\mathcal{DG}^1 \cap \mathcal{DG}^2)) \).

Proof: According to the proofs of Theorem 2 and Theorem 3, the minimum total capacity required on link \( l \) to accommodate all working paths on link \( l \) and to protect the failure of link \( k \) only depends on the demands in \( \mathcal{DG}^0 \) and \( \mathcal{DG}^1 \cap \mathcal{DG}^2 \). It is obvious that \( \mathcal{DG}^0 \) and \( \mathcal{DG}^1 \cap \mathcal{DG}^2 \) do not have common demands because the working path and protection path of a demand must be link disjoint, i.e., \( \mathcal{DG}^0 \cap (\mathcal{DG}^1 \cap \mathcal{DG}^2) = \emptyset \). Similar to the proofs of Theorem 2 and Theorem 3, we conclude that \( g_{lk} = \omega(\mathcal{DG}^0 \cup (\mathcal{DG}^1 \cap \mathcal{DG}^2)) \).

As a result, the capacity provision matrix \( G = \{ g_{lk} \}_{L \times L} = \{ \omega(\mathcal{DG}^0 \cup (\mathcal{DG}^1 \cap \mathcal{DG}^2)) \}_{L \times L} \). \( \mathcal{DG}^0 \) and \( \mathcal{DG}^1 \cap \mathcal{DG}^2 \) are determined in Eqn. (2.74) and Eqn. (2.75), respectively:

\[
\mathcal{DG}^0 = \{ r | a_{rl} = 1, \ r \in \mathcal{D} \}, \tag{2.74}
\]

\[
\mathcal{DG}^1 \cap \mathcal{DG}^2 = \{ r | b_{rl} = 1 \text{ and } a_{rk} = 1, \ r \in \mathcal{D} \}. \tag{2.75}
\]

To determine matrix \( G \), we therefore need to know matrices \( A \) and \( B \), as well as the capacity requirements of the demands in the given demand set \( \mathcal{D} \) and the time.
correlation among them. Hereinafter we will use $G = \Pi(D, A, B)$ to represent the calculation of the capacity provision matrix.

### 2.3.3 Demand Ordering Policies

Given a set of scheduled traffic demands $D$, the proposed ISR algorithm first sorts the demands based on a chosen ordering policy, which results in a scheduling order of demands. Different ordering policies are proposed below and their impacts will be studied. The ordered demands are saved in a new set $D'$ and are processed sequentially.

**Earliest-Setup Demand First (ESDF)** This policy schedules demands in increasing order of their setup times, i.e., the earlier a demand starts, the earlier it will be scheduled. For demands with the same setup time, the demand with an earlier teardown time will be scheduled first.

**Earliest-Teardown Demand First (ETDF)** This policy schedules demands in increasing order of their teardown times, i.e., the earlier a demand ends, the earlier it will be scheduled. For demands with the same teardown time, the demand with an earlier setup time will be scheduled first.

**Most Conflicting Demand First (MCDF)** Given a traffic demand $r \in D$, we define $T(r)$ as the set of demands in $D$ which overlap with $r$ in time, i.e.,

$$T(r) = \{ r' \mid r \text{ and } r' \text{ overlap in time}, \ r' \in D, r' \neq r \}.$$  

(2.76)

$|T(r)|$ represents the number of demands that overlaps with demand $r$ in time and is termed as the time conflict index of demand $r$.

This policy schedules demands in decreasing order of their time conflict indices. That is, the more a demand overlaps in time with other demands in $D$, the earlier it will be processed. For demands with the same time conflict index, policies *ESDF* and *ETDF* will be applied in order.
**Least Conflicting Demand First (LCDF)** In contrast to **MCDF**, this policy schedules demands in increasing order of their time conflict indices. For demands with the same time conflict index, policies **ESDF** and **ETDF** will be applied in order.

### 2.3.4 Iterative Survivable Routing

In this study, we assume non-bifurcated routing. For demand \( r \) to be processed next that requires \( n_r \) lightpaths, our objective is to find a pair of link-disjoint paths such that there is only one physical route for either path and \( n_r \) wavelengths will be used on all the links along each path.

The proposed **Iterative Survivable Routing** algorithm finds a pair of paths for each demand in \( D' \) iteratively as follows after initialization (Step 1 of Fig. 2.6). For each demand \( r \in D' \), the algorithm first finds its working path (Step 3 of Fig. 2.6); and then finds its protection path (Step 4 of Fig. 2.6). The routing of working path and protection path of an individual demand \( r \) is conducted by the **Single Demand Routing** (SDR) procedure (given in Fig. 2.7). This completes one iteration of the ISR algorithm. The details of the **Single Demand Routing** procedure will be elaborated on shortly. After accommodating all the demands in \( D' \) (i.e., working paths and protection paths of all the demands have been found), for each demand \( r \in D' \), the **Iterative Survivable Routing** algorithm, in the next iteration, tries to find a new working path (and/or a new protection path) which consumes less capacity than the path used by demand \( r \) in the previous iteration. If a better path is returned by the SDR procedure, this path replaces the old path used by \( r \). Let \( t^A \) and \( t^B \) be the number of times that an old working path and an old protection path are replaced during an iteration, respectively. The ISR algorithm goes to a new iteration unless no path is replaced in an iteration of execution (Step 5 of Fig. 2.6).

We now describe how the working path and protection path of individual demands are routed using the **Single Demand Routing** procedure. Fig. 2.7 shows the **Single
Iterative Survivable Routing\((N, L, K, D')\)

1. Initialize \(A = B = 0_{D \times L}, G = 0_{L \times L}\);
2. \(t^A = t^B = 0\);
3. For each demand \(r \in D'\) do
   \(\text{Single Demand Routing}(N, L, K, r, WP)\);
   Increase \(t^A\) by one if working path of \(r\) is updated;
4. For each demand \(r \in D'\) do
   \(\text{Single Demand Routing}(N, L, K, r, PP)\);
   Increase \(t^B\) by one if protection path of \(r\) is updated;
5. If \(t^A + t^B \geq 1\), go to step 2; exit otherwise;

\[\text{Fig. 2.6: Iterative Survivable Routing algorithm.}\]

Demand Routing (SDR) algorithm that finds the working path for demand \(r\). The algorithm for finding the protection path for demand \(r\) can be similarly obtained by replacing \(a_r\) and \(A\) with \(b_r\) and \(B\), respectively, in the algorithm. Step 1 of the SDR algorithm calculates the total capacity, \(\Lambda\), used by all demands accommodated in the network so far using Eqn. (2.73). To find a better working path for demand \(r\), the SDR algorithm calculates, in Step 2, a new vector of link metrics \(c_r\), based on which a path is then calculated as the potential new working path for demand \(r\). The calculation of link metrics is explained in greater details as follows:

**C1** The current working path \(a_r\) of demand \(r\) is removed from the network to obtain the new working path link incidence matrix \(A^0\), i.e., \(A^0 = A\) except that \(a_r = 0_{1 \times L}\) in \(A^0\). Based on the new \(A^0\), calculate \(G^0 = \Pi(D, A^0, B)\) and \(s^0 = \max G^0\) using **Theorem 4** and Eqn. (2.71) to obtain the capacity provision matrix and the minimum total capacity required on each link.

**C2** Let \(a^*_r = e^T - b_r\) denote all eligible links that can be used by the working path of demand \(r\) after all links used by its corresponding protection path are removed. Assuming these eligible links are used to route the working path of demand \(r\), the new working path link incidence matrix \(A^* = A\) except that \(a_r = a^*_r\) in \(A^*\). The capacity provision matrix is \(G^* = \Pi(D, A^*, B)\); and the vector of minimum total link capacity required is \(s^* = \max G^*\).
The vector of link metrics for demand \( r \) is obtained as

\[
c_r = \{c_{rl}\}_{L \times 1} = s^* - s^0.
\] (2.77)

The element \( c_{rl} \) of \( c_r \) indicates the incremental capacity needed on link \( l \) if this link is used by the working path of demand \( r \).

After obtaining the vector of link metrics, Step 3 of the SDR algorithm first excludes all the links used by the protection path \( b_r \); the algorithm then runs a shortest path algorithm (e.g., Dijkstra’s algorithm) based on link metrics \( c_r \) to find a new working path \( \rho \) for demand \( r \). The temporary new working path \( \rho \) is used to update the working path link incidence matrix \( A_{\text{new}} = A \) except that \( a_r = \rho \) in \( A_{\text{new}} \). Then in Step 4, the new capacity provision matrix and the new vector of minimum required link capacity are recalculated as \( G_{\text{new}} = \Pi(D, A_{\text{new}}, B) \) and \( s_{\text{new}} = \max G_{\text{new}} \), respectively. Subsequently, the new total required network capacity for all demands including demand \( r \) is obtained, i.e., \( \Lambda_{\text{new}} = e^T s_{\text{new}} \), if demand \( r \) is routed on the new path \( \rho \).

In the final step, the new working path \( \rho \) replaces the original working path \( a_r \) if \( \rho \) is the very first working path of demand \( r \), i.e., \( a_r = 0_{1 \times L} \); or \( \Lambda_{\text{new}} < \Lambda \), i.e., the new working path results in less total capacity used. That is, the newly calculated working path is committed if the following condition holds

\[
a_r = \rho, \text{ if } a_r = 0_{1 \times L} \text{ or } \Lambda_{\text{new}} < \Lambda.
\] (2.78)

The path link incidence matrix \( A \), the capacity provision matrix \( G \), and the vector \( s \) will also be updated to reflect the working path change accordingly.

### 2.3.5 Numerical and Simulation Results

We present and evaluate the performance of the proposed ISR algorithm and compare it against that of ILP4 and ILP8 since shared path protection under the scheduled
Single Demand Routing ($\mathcal{N}, \mathcal{L}, \mathcal{K}, r, \text{flag}$)
1. Calculate current total capacity used $\Lambda$ using Eqn. (2.73);
2. Calculate link metrics $c_r$ according to $C1-C3$;
3. Find shortest path $\rho$ using $c_r$;
4. Calculate the new total used capacity $\Lambda^{\text{new}}$ using Eqn. (2.73);
5. If $\Lambda^{\text{new}} < \Lambda$ or $a_r = 0_{1 \times L}$ (when flag is WP) $\Lambda$
   Update $A$ with path $\rho$, $G$ and $s$.

Fig. 2.7: Single demand routing algorithm for finding a working path of demand $r$.

Traffic model is employed in these two optimization models as well as the ISR algorithm. The objective of all the optimization models and the ISR algorithm is to determine a pair of working path and protection path, and assign wavelengths to them for each demand in a scheduled traffic demand set, so that the total number of wavelength-links used is minimized provided that network resources are sufficient to accommodate all the demands.

We first evaluate the performance of the ISR algorithm and compare it against that of ILP4 and ILP8. Then, we will study and compare the performances of the ISR algorithm with various demand ordering policies in larger networks with larger demand sets.

Iterative Survivable Routing Algorithm versus ILPs

Table 2.3 shows the total number of wavelength-links required to accommodate different traffic demand sets in ILP4, ILP8 and the ISR algorithm with different demand ordering polices when the demand time correlation are weak, medium, and strong, respectively. From the table, we observe that in the first two scenarios, the size of example networks is small; the set of traffic demands is small; and the number of wavelengths on each link is not large. In these scenarios, the optimization models ILP4 and ILP8 are solved with the optimal solutions. ILP8 is also solved with the optimal solutions in Scenario 3. In larger networks with large demand sets, however, ILP4 and ILP8 cannot be solved with the optimal solutions within the time limit, even though ILP8 employs two-step optimization approach, as shown in the last
three scenarios in Table 2.3.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>τ</th>
<th>ILP4</th>
<th>ILP8</th>
<th>ISR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ESDF</td>
</tr>
<tr>
<td>1</td>
<td>W</td>
<td>11*</td>
<td>11*</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>12*</td>
<td>12*</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>13*</td>
<td>13*</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>W</td>
<td>24*</td>
<td>24*</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>30*</td>
<td>30*</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>33*</td>
<td>34*</td>
<td>35</td>
</tr>
<tr>
<td>3</td>
<td>W</td>
<td>43</td>
<td>44*</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>54</td>
<td>53*</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>68</td>
<td>66*</td>
<td>65</td>
</tr>
<tr>
<td>4</td>
<td>W</td>
<td>78</td>
<td>81</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>91</td>
<td>91</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>125</td>
<td>124</td>
<td>112</td>
</tr>
<tr>
<td>5</td>
<td>W</td>
<td>92</td>
<td>90</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>104</td>
<td>112</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>130</td>
<td>124</td>
<td>104</td>
</tr>
<tr>
<td>6</td>
<td>W</td>
<td>119</td>
<td>120</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>152</td>
<td>136</td>
<td>119</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>218</td>
<td>182</td>
<td>160</td>
</tr>
</tbody>
</table>

Tab. 2.3: Total number of wavelength-links used. * indicates the optimal solution found and W, M, S represent weak, medium and strong time correlation, respectively. τ is the time correlation of a demand set.

In the first three scenarios in Table 2.3, we observe that the ISR algorithm (with four different demand ordering policies) achieves the same results in almost all cases; and the results are very close to that of ILP4 and ILP8, or even better than that of ILP4 in Scenario 3 in which ILP4 only obtains sub-optimal solutions within the time limit. As the network size, demand set size or wavelength set size increases, the computational complexity of ILP4 and ILP8 prevent them from achieving good solutions within the time limit. In contrast, the ISR algorithm achieves much better performance in much less time, as shown in Scenarios 4, 5 and 6. In addition, we observe that the performances of the ISR algorithm employing various ordering policies do not differ significantly in all the scenarios investigated.

Table 2.4 shows the computational time of ILP4, ILP8 and the ISR algorithm when the demand time correlation are weak, medium, and strong, respectively, in the 6 scenarios investigated. A time range is given if different demand scheduling policies
in the ISR algorithm result in different computational time. The results in the table indicate that the proposed ISR algorithm is extremely time efficient (several orders of magnitude less) while achieving excellent performance in terms of total network resources used, which has been shown in Table 2.3.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>τ</th>
<th>ILP4</th>
<th>ILP8</th>
<th>ISR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>W</td>
<td>1s</td>
<td>&lt; 1s</td>
<td>(2.0 − 2.1) × 10⁻⁴s</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>5s</td>
<td>&lt; 1s</td>
<td>(2.3 − 2.4) × 10⁻⁴s</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>10s</td>
<td>&lt; 1s</td>
<td>(2.3 − 2.4) × 10⁻⁴s</td>
</tr>
<tr>
<td>2</td>
<td>W</td>
<td>4s</td>
<td>1s</td>
<td>(1.1 − 1.6) × 10⁻³s</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>4m</td>
<td>5s</td>
<td>(2.7 − 2.8) × 10⁻²s</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>2h10m</td>
<td>20s</td>
<td>(1.3 − 1.7) × 10⁻³s</td>
</tr>
<tr>
<td>3</td>
<td>W</td>
<td>×</td>
<td>17s</td>
<td>2.3 × 10⁻²s</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>×</td>
<td>1h20m</td>
<td>3.0 × 10⁻²s</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>×</td>
<td>3h14m</td>
<td>4.1 × 10⁻²s</td>
</tr>
<tr>
<td>4</td>
<td>W</td>
<td>×</td>
<td>×</td>
<td>(0.10 − 0.11)s</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>×</td>
<td>×</td>
<td>(0.12 − 0.13)s</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>×</td>
<td>×</td>
<td>(0.13 − 0.14)s</td>
</tr>
<tr>
<td>5</td>
<td>W</td>
<td>×</td>
<td>×</td>
<td>(0.15 − 0.22)s</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>×</td>
<td>×</td>
<td>(0.22 − 0.27)s</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>×</td>
<td>×</td>
<td>(0.24 − 0.27)s</td>
</tr>
<tr>
<td>6</td>
<td>W</td>
<td>×</td>
<td>×</td>
<td>(0.95 − 1.23)s</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>×</td>
<td>×</td>
<td>(1.33 − 1.80)s</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>×</td>
<td>×</td>
<td>(1.40 − 1.68)s</td>
</tr>
</tbody>
</table>

Tab. 2.4: Computational time. × indicates the 4-hour time limit. h, m and s represent hour(s), minute(s) and second(s), respectively. τ is the time correlation of a demand set.

Impact of Ordering Policies on Iterative Survivable Routing Algorithm in Larger Networks

In this subsection, we investigate the performance of the ISR algorithm using demand sets of various sizes in the NSFNET topology. Figs. 2.8 (a), (b) and (c) show the total number of wavelength-links required by the ISR algorithm versus the number of demands in a demand set. The ISR algorithm uses different demand scheduling policies and is applied to demand sets with different demand time correlations (weak, medium, and strong). Although the performances resulting from the four policies appear to be close, we observe that the ISR algorithm employing the Most Conflicting Demand First (MCDF) policy achieves the best performance in most scenarios, and the one using the LCDF policy needs more wavelength-links than the three other policies.
2.4 Heuristic Algorithm with SRLG

In this section, we extend our heuristic algorithm design described in Chapter 2.3 from the single link failure model to the single SRLG failure model so that 100% restorability is guaranteed against any single SRLG failure in a WDM optical network.

2.4.1 SRLG Failure Model

We assume that there exist $F$ SRLGs in the network. We use a binary matrix $\mathbf{H} = \{h_f\}_{F \times 1} = \{h_{fl}\}_{F \times L}$ to characterize these SRLGs. The row vector $\mathbf{h}_f$ in $\mathbf{H}$ is for SRLG $f$ and its element $h_{fl}$ takes on 1 if link $l$ fails in SRLG $f$. In this way, each SRLG failure is associated with a set of links that will fail simultaneously in the failure scenario. Let the set of links be $\mathcal{S}(f)$. Moreover, we use two binary matrices $\mathbf{P}^A = \{p^A_r\}_{D \times 1} = \{p^A_{rf}\}_{D \times F}$ and $\mathbf{P}^B = \{p^B_r\}_{D \times 1} = \{p^B_{rf}\}_{D \times F}$ to denote the SRLG failure matrix for the working paths and protection paths of all traffic demands,
respectively, where $p_A^{rf}/p_B^{rf} = 1$ if the working/protection path of demand $r$ will be affected by SRLG $f$, and $p_A^{rf}/p_B^{rf} = 0$ otherwise. In other words, the row vector $p_A^{rf}/p_B^{rf}$ indicates the SRLGs through which the working (protection) path of demand $r$ passes. Matrices $P^A$ and $P^B$ can be calculated using Eq. (2.79) and Eq. (2.80), respectively. Here $A$ and $B$ represent the working path link incidence matrix and protection path link incidence matrix, respectively, as described in Chapter 2.3. Note that a binary matrix multiplication operation “$\odot$” is used in these two equations. It modifies the general addition in $1 + 1 = 2$ to Boolean addition $1 + 1 = 1$ [68]. By using this binary operation, the complicated logical relations among paths, links and SRLG failure scenarios are simplified into one matrix operation.

$$P^A = A \odot H^T$$  \hspace{1cm} (2.79)

$$P^B = B \odot H^T$$  \hspace{1cm} (2.80)

In the link failure model, the working path and the protection path of the same traffic demand should not share a physical link in order to achieve 100% restorability. We call the links that are used by one working path tabu-links of its corresponding protection path, and vice versa. In the single SRLG failure model, similarly, the working path and the protection path of the same demand must use links that belong to different SRLGs. More specially, the set of tabu-links of a working (protection) path includes all the links that belong to the SRLGs through which its corresponding protection (working) path passes. We use two binary matrices $T^A = \{t^A_r\}D \times 1 = \{t^A_{rl}\}D \times L$ and $T^B = \{t^B_r\}D \times 1 = \{t^B_{rl}\}D \times L$ to denote the tabu-link matrices of the working paths and protection paths of all traffic demands, respectively, where $t^A_{rl}/t^B_{rl} = 1$ if the working (protection) path of demand $r$ is not allowed to use link $l$, and $t^A_{rl}/t^B_{rl} = 0$ otherwise. We use the operation “$\odot$” again to calculate $T^A$ and $T^B$ in Eq. (2.81) and Eq. (2.82), respectively.

$$T^A = P^B \odot H$$  \hspace{1cm} (2.81)
\[ T^B = P^A \odot H \] (2.82)

In the SRLG failure model, we re-define the capacity provision matrix as \( G = \{g_{lf}\}_{L \times F} \) whose element \( g_{lf} \) indicates the minimum total capacity required on link \( l \) to accommodate all working paths on link \( l \) and protect against the failure of SRLG \( f \). We use the minimum total capacity vector \( s \) and \( \Lambda \) (i.e., the total number of wavelength-links used by working paths and protection paths of all demands) as defined in Chapter 2.3.

We use an example to further explain the notation discussed above. Two static demands are routed in the network of Fig. 2.9. The figure illustrates the capacity provision matrix \( G \), the vector of minimum required capacity of links \( s \), and the total number of wavelength-links \( \Lambda \) used to accommodate the two demands. The working path and protection path of demand 1 between nodes 0 and 1 are \( WP1 \) \((0 \rightarrow 1) \) and \( PP1 \) \((0 \rightarrow 4 \rightarrow 5 \rightarrow 2 \rightarrow 1) \). The working path and protection path of demand 2 between nodes 0 and 2 are \( WP2 \) \((0 \rightarrow 1 \rightarrow 2) \) and \( PP2 \) \((0 \rightarrow 4 \rightarrow 5 \rightarrow 2) \). Only seven links in the network are used by the paths of the two demands and they only go through six SRLGs. Therefore, only these links and SRLGs are included in the matrices to reduce the size of the matrices in the example. The numbers besides the links (but not in the parentheses) are used to number the links while the numbers in the parentheses indicate the corresponding SRLG that the link belong to. Path link incidence matrices \( A \) and \( B \) give the working paths and protection paths used by the two demands. Matrix \( H \) shows the set of links that will fail simultaneously in an SRLG failure scenario. Notice that in this example, links 1 and 4 belong to SRLG 1 and all other links belong to a SRLG that only includes a single link. Matrices \( P^A \), \( P^B \), \( T^A \) and \( T^B \) are then calculated based on \( A \), \( B \) and \( H \). The calculation of the capacity provision matrix \( G \) is not straightforward and will be explained in detail below. After obtaining matrix \( G \), \( s \) and \( \Lambda \) can be easily obtained.
The minimum total capacity required on link \( l \) to accommodate all working paths on link \( l \) and to protect against the failure of SRLG \( f \) is equal to the weight of the maximum weight clique of the weighted interval graph corresponding to the demand group in addition to the demand groups \( D \mathcal{G}^0 \) and \( D \mathcal{G}^1 \) defined before. Fig. 2.10 shows these three demand groups with a simplified network.

- **Demand Group 3:** \( D \mathcal{G}^2 = \{ r^3_1, r^3_2, ..., r^3_{g_3} \} \) where \( r^3_i (1 \leq i \leq g_3) \) is a demand whose working path passes through a link in \( \mathcal{S}(f) \), i.e., the set of links that will fail simultaneously in SRLG \( f \).

Theorem 5: The minimum total capacity required on link \( l \) to accommodate all working paths on link \( l \) and to protect against the failure of SRLG \( f \) is equal to the weight of the maximum weight clique of the weighted interval graph corresponding to the demand group in addition to the demand groups \( D \mathcal{G}^0 \) and \( D \mathcal{G}^1 \) defined before. Fig. 2.10 shows these three demand groups with a simplified network.

- **Demand Group 3:** \( D \mathcal{G}^2 = \{ r^3_1, r^3_2, ..., r^3_{g_3} \} \) where \( r^3_i (1 \leq i \leq g_3) \) is a demand whose working path passes through a link in \( \mathcal{S}(f) \), i.e., the set of links that will fail simultaneously in SRLG \( f \).
mand set which includes the demands in Demand Group 0 and those in both Demand Group 1 and Demand Group 3. That is, \( g_{l,f} = \omega(\mathcal{DG}^0 \cup (\mathcal{DG}^1 \cap \mathcal{DG}^3)) \).

**Proof:** When SRLG \( f \) fails, only the protection paths on link \( l \) whose corresponding working paths pass through any link in \( S(f) \) need capacity assignment to protect the failure because only those working traffic will be affected and need to be protected on link \( l \). The set of involved demands includes the demands in both Demand Group 1 and Demand Group 3, i.e., \( \mathcal{DG}^1 \cap \mathcal{DG}^3 \). Let \( \nu_{l,f} \) be the minimum capacity required by protection paths on link \( l \) when SRLG \( f \) fails. Similar to the proof of *Theorem 3*, therefore, \( \nu_{l,f} = \omega(\mathcal{DG}^1 \cap \mathcal{DG}^3) \). In order to accommodate all working paths on link \( l \) as well, \( g_{l,f} \) depends on the demands in \( \mathcal{DG}^1 \cap \mathcal{DG}^3 \) and the demands in \( \mathcal{DG}^0 \). Similar to the proof of *Theorem 4*, finally, the minimum total capacity required on link \( l \) is \( g_{l,f} = \omega(\mathcal{DG}^0 \cup (\mathcal{DG}^1 \cap \mathcal{DG}^3)) \). \( \Box \)

In the single SRLG failure model, the capacity provision matrix becomes \( G = \{g_{l,f}\}_{L \times F} = \{\omega(\mathcal{DG}^0 \cup (\mathcal{DG}^1 \cap \mathcal{DG}^3))\}_{L \times F} \). \( \mathcal{DG}^0 \) is determined in Eq. (2.74), and \( \mathcal{DG}^1 \cap \mathcal{DG}^3 \) can be determined in Eq. (2.83). To determine the whole matrix \( G \), we therefore need to know matrix \( P^A \) as well as the information needed to calculate \( G \) in the single link failure model (i.e., the matrices \( A, B \) and the original demand set \( D \)).
Thereinafter we use \( G = \Pi(\mathcal{D}, A, B, P^A) \) to represent the calculation of the capacity provision matrix.

\[
\mathcal{DG}^1 \bigcap \mathcal{DG}^3 = \{ r | b_{rf} = 1 \text{ and } p_{rf}^A = 1, \ r \in \mathcal{D} \} 
\]  
(2.83)

2.4.2 Iterative Survivable Routing in SRLG Failure Model: ISR-SRLG

We use an iterative algorithm which is similar to ISR, called ISR-SRLG, to process demands in the single SRLG failure model. For demand \( r \) to be processed next that requires \( n_r \) lightpaths, our objective is to find a pair of SRLG-disjoint paths such that there is only one physical route for either path and \( n_r \) wavelengths will be used on all the links along each path.

Given a set of scheduled traffic demands \( \mathcal{D} \), ISR-SRLG first employs the Most Conflicting Demand First policy to sort the demands and saves the ordered demands in a new set \( \mathcal{D}' \). The Most Conflicting Demand First policy achieves the best performance in most scenarios among different demand ordering policies studied in Chapter 2.3.

The ISR-SRLG algorithm processes demands in \( \mathcal{D}' \) sequentially. It finds a pair of paths for each demand iteratively as follows after initialization (Step 1 of Fig. 2.11). For each demand \( r \in \mathcal{D}' \), the algorithm first finds its working path (Step 3 of Fig. 2.11); and then finds its protection path (Step 4 of Fig. 2.11). The routing of working path and protection path of an individual demand \( r \) is conducted by the Single Demand Routing with SRLG (SDR-SRLG) procedure (given in Fig. 2.12). This completes one iteration of the ISR-SRLG algorithm. The details of the SDR-SRLG procedure will be elaborated on shortly. After accommodating all the demands in \( \mathcal{D}' \) (i.e., working paths and protection paths of all the demands have been found), for each demand \( r \in \mathcal{D}' \), the ISR-SRLG algorithm, in the next iteration, tries to find a new working path (and/or a new protection path) which consumes less capacity than the path used by demand \( r \) in the previous iteration. If a better path is returned by the SDR-SRLG procedure, this path replaces the old path used by \( r \). Let \( t^A \) and
be the number of times that an old working path and an old protection path are replaced during an iteration, respectively. The ISR-SRLG algorithm goes to a new iteration unless no path is replaced in an iteration of execution (Step 5 of Fig. 2.11).

Iterative Survivable Routing with SRLG \((N, L, K, D')\)

1. Initialize \(A = B = 0_{D \times L}, P^A = P^B = 0_{D \times F}, T^A = T^B = 0_{D \times L}, G = 0_{L \times F};\)
2. \(t^A = t^B = 0;\)
3. For each demand \(r \in D'\) do
   - Single Demand Routing with SRLG \((N, L, K, r, WP);\)
   - Increase \(t^A\) by one if working path of \(r\) is updated;
4. For each demand \(r \in D'\) do
   - Single Demand Routing with SRLG \((N, L, K, r, PP);\)
   - Increase \(t^B\) by one if protection path of \(r\) is updated;
5. If \(t^A + t^B \geq 1\), go to step 2; exit otherwise;

**Fig. 2.11:** Iterative Survivable Routing algorithm under the single SRLG failure model.

We now describe how the working path and protection path of individual demands are routed using the SDR-SRLG procedure. Fig. 2.12 shows the Single Demand Routing with SRLG (SDR-SRLG) algorithm that finds the working path for demand \(r\). The algorithm of finding a protection path can be easily obtained by replacing \(a_r, A, p^A_r\) and \(t^B_r\) with \(b_r, B, p^B_r\) and \(t^A_r\), respectively, in the steps in Fig. 2.12. Step 1 of the SDR-SRLG algorithm calculates the total capacity, \(\Lambda\), used by all demands accommodated in the network so far using Eq. (2.73). To find a better working path for demand \(r\), the SDR-SRLG algorithm calculates, in Step 2, a new vector of link metrics \(c_r\), based on which a path is then calculated as the potential new working path for demand \(r\). The calculation of link metrics is explained in greater detail as follows:

**C1** The current working path \(a_r\) of demand \(r\) is removed from the network to obtain the new working path link incidence matrix \(A^0\), i.e., \(A^0 = A\) except that \(a_r = 0_{1 \times L}\) in \(A^0\). Based on the new \(A^0\), calculate \(G^0 = \Pi(D, A^0, B, P^A)\) and \(s^0 = \max G^0\) using Theorem 4 and Eq. (2.71) to obtain the capacity provision matrix and the minimum total capacity required on each link.
Let $a_r^* = e^T - t_r^A$ denote all eligible links that can be used by the working path of demand $r$ after all links that share the same SRLG with its corresponding protection path are removed. Assuming these eligible links are used to route the working path of demand $r$, the new working path link incidence matrix $A^* = A$ except that $a_r = a_r^*$ in $A^*$. The capacity provision matrix is $G^* = \Pi(D, A^*, B, P^A)$; and the vector of minimum total link capacity required is $s^* = \max G^*$.

C3 The vector of link metrics for demand $r$ is obtained as

$$c_r = \{c_{rl}\}_{L \times 1} = s^* - s^0. \tag{2.84}$$

The element $c_{rl}$ of $c_r$ indicates the incremental capacity needed on link $l$ if this link is used by the working path of demand $r$.

In step 3, all the tabu-links of $a_r$, which are recorded in $t_r^A$, are removed, and a shortest path algorithm (e.g., Dijkstra’s algorithm) based on link metrics $c_r$ is then used to find a new working path $\rho$. In Step 4, the algorithm calculates $G^{new} = \Pi(D, A^{new}, B, P^A)$ and $s^{new} = \max G^{new}$ where $A^{new} = A$ except that $a_r = \rho$ in $A^{new}$. Then the algorithm obtains $\Lambda^{new} = e^T s^{new}$. In the final step, the algorithm replaces the original working path $a_r$ with the new path $\rho$ if demand $r$ does not have a working path (i.e., $a_r = 0_{1 \times L}$) or $\Lambda^{new} < \Lambda$, i.e., the new path results in less total capacity, as shown in Eq. (2.85). The algorithm then updates $p_r^A$ and $t_r^B$ in Eq. (2.86) and Eq. (2.87), respectively. Note that the updated $t_r^B$ will be used when finding the protection path of demand $r$ in the same iteration. The capacity provision matrix $G$ and capacity vector $s$ will also be updated to reflect the change of working path accordingly.

$$a_r = \rho, \text{ if } a_r = 0_{1 \times L} \text{ or } \Lambda^{new} < \Lambda. \tag{2.85}$$

$$p_r^A = \rho \odot H^T \tag{2.86}$$

$$60$$
\[ t_r^B = p_r^A \odot H \] (2.87)

**Single Demand Routing with SRLG** \((N, L, K, r, \text{flag})\)

1. Calculate current total capacity used \(\Lambda\) using Eq. (2.73);
2. Calculate link metrics \(c_r\) according to C1-C3;
3. Find shortest path \(\rho\) using \(c_r\);
4. Calculate the new total used capacity \(\Lambda_{\text{new}}\) using Eq. (2.73);
5. If \(\Lambda_{\text{new}} < \Lambda\) or \(a_r = 0_{1 \times L}\) (when flag is WP)
   - Update \(A\) with path \(\rho\), \(p_r^A\), \(t_r^B\), \(G\) and \(s\).

**Fig. 2.12:** Single demand routing algorithm for finding a working path under the single SRLG failure model.

### 2.4.3 Numerical and Simulation Results

In this section, we evaluate the performance of the proposed ISR-SRLG algorithm described in the previous section. The objective of the algorithm is to determine a pair of SRLG-disjoint working path and protection path, and assign wavelengths to them for each demand in a scheduled traffic demand set, so that the total number of wavelength-links used is minimized provided that network resources are sufficient to accommodate all the demands. In the simulation study, however, we limit the number of wavelengths per link to be 100. We use the same network topology and same simulation settings as those used in Chapter 2.3.

**SRLGs Settings and Generation**

SRLGs are known a priori for the survivable service provisioning problem. In our simulation study, they are randomly generated. To facilitate this, we define the term \(\gamma\)-SRLG set where \(\gamma\) is a positive integer. A \(\gamma\)-SRLG set has the property that it contains \(\gamma\) links. That is, the size of a \(\gamma\)-SRLG set is \(\gamma\). Hereinafter, we call a network a \(\gamma\)-SRLG network if the maximum size of all SRLG sets in the network is \(\gamma\). For a \(\gamma\)-SRLG network, we randomly generate SRLGs according to the following rules for each SRLG:
• Any single SRLG failure does not disconnect the network;

• Each link in the network belongs to at least one SRLG. That is, there is no risk-free link in the network;

• Each SRLG set contains 1 to $\gamma$ links.

Note that each link in a network with single link failure can be considered as a 1-SRLG set and thus the single link failure model is a special case of the single SRLG failure model.

For a network with a SRLG size constraint, we generate SRLGs using two methods. The first method randomly picks a node, and then randomly picks the links that are within $h$ hops from the selected node to be included in an SRLG. The method produces $h$-hop localized SRLGs. The second method randomly picks links in the topology to be included in an SRLG. This method produces non-localized SRLGs. We use the NSFNET topology in our simulations and set $h$ to be 3.

**Heuristic Algorithm Performance: Different SRLG Sizes**

Figures 2.13, 2.14 and 2.15 show the total number of wavelength-links required to accommodate different traffic demand sets in the ISR-SRLG algorithm with different SRLG set sizes when the demand set time correlation are weak, medium, and strong, respectively. As expected, we observe that as the size of SRLGs increases, more wavelength-links are needed to accommodate the same traffic demand set. One reason could be when an SRLG contains more links, the probability of a path passing through links that belong to the same SRLG gets higher. This may force the path to use a longer route. Another reason could be that more working capacity will be affected by a single SRLG failure as an SRLG becomes larger. In this case, two working paths are easier to be SRLG-joint and thus their corresponding protection paths cannot share any wavelength-link. This lowers the degree of network resource sharing.

We also observe that when the size of SRLGs and/or the size of demand set
become large enough (e.g., $\gamma = 3$ and $|D| \geq 30$), the traffic demand set cannot be accommodated based on the limited network resources (100 wavelengths per link), i.e., the algorithm is unable to find a working path and a protection path for all the demands in $D$ that satisfy the shared path based protection constraints and provide 100% restorability for all demands to protect against a single SRLG failure. This is true whether the SRLG sets are localized or not, or whether the demand time correlation is strong or weak.

Fig. 2.13: Comparison of SRLG size: Total number of wavelength-links versus number of demands under weak time correlation.

Fig. 2.14: Comparison of SRLG size: Total number of wavelength-links versus number of demands under medium time correlation.

Heuristic Algorithm Performance: Different Types of SRLGs
Figure 2.16 depicts the total number of wavelength-links required in the network with localized SRLGs or non-localized SRLGs when the demand time correlation are weak, medium, and strong, respectively. Generally, the performance resulting from the two SRLG generation methods (i.e., localized or non-localized) appears to be very close in the figures for the cases we studied.

In all the cases we studied, our proposed algorithm is very time efficient and the execution time roughly ranges from half a second to half a minute as the number of demands increases from 10 to 100.

2.5 Summary

In this chapter, we have developed various optimization models and efficient heuristic algorithms to solve the survivability provisioning problem in WDM optical networks under the scheduled traffic model. We have developed two joint RWA ILP formulations, ILP1 and ILP2, for dedicated and shared protection schemes in survivable WDM optical mesh networks under the conventional static traffic model and two joint RWA ILP formulations, ILP3 and ILP4, under the scheduled traffic model. To solve large problems, we separate the joint RWA problem into a routing subproblem and a wavelength assignment subproblem. For the routing subproblem, we use ILP5 to
pre-compute a pair of link-disjoint routes for each of the demands in the traffic demand set $\mathcal{D}$ as its working path and protection path. For the wavelength assignment subproblems, the pair of routes for each demand $r$ in $\mathcal{D}$ is assigned proper wavelengths such that the total network resources (i.e., the number of wavelength-links) used by all the demands are minimized. ILP6, ILP7 and ILP8 have been developed in the wavelength assignment subproblem for protection schemes SP, DP-S and SP-S, respectively. The simulation results showed that the protection schemes under the scheduled traffic model (i.e., DP-S and SP-S) use much fewer wavelength-links than their counterparts under the conventional static traffic model (i.e., DP and SP). The reason is that DP-S and SP-S are able to exploit the time-disjointness among

---

Fig. 2.16: Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links used in 2-SRLG NSFNET network.
demands and reuse wavelength-links as much as possible.

We have proposed a capacity provision matrix based Iterative Survivable Routing (ISR) algorithm to solve the survivable service provisioning problem under the scheduled traffic model. We first order the scheduled demands based on different demand ordering policies and then process demands sequentially. The algorithm runs iteratively to approximate the optimal solution. The simulation results showed that the ISR can achieve very good performance that is very close to the optimal solutions obtained by ILP4 and ILP8 when the size of example networks is small, the set of traffic demands is small, and the number of wavelengths on each link is not large. As the network size, demand set size or wavelength set size increases, the computational complexity of ILP4 and ILP8 prevent them from achieving good solutions within the time limit. In contrast, the ISR algorithm achieves much better performance in much less time. In addition, we have observed that the performances of the ISR algorithm employing various scheduling policies do not differ significantly in all the scenarios investigated.

We have also extended the heuristic algorithm design from the single link failure model to the single SRLG failure model. The extended heuristic algorithm ISR-SRLG is able to solve the survivable routing problem in WDM optical networks in which 100% restorability is guaranteed against any single SRLG failure under the scheduled traffic model. We consider both localized SRLGs and non-localized SRLGs, as well as SRLG sets with different sizes. The simulation results showed that as the size of SRLG set increases, more wavelength-links are needed to accommodate the same traffic demand set. When the size of SRLG sets and/or the size of demand set become large enough, the traffic demand set could not be accommodated based on the limited network resources. The results also showed that, in general, the performances in terms of total number of wavelength-links required in the network resulting from the two SRLG generation methods (i.e., localized or non-localized) appear to be very close.
for the cases we studied.
3. SLIDING SCHEDULED TRAFFIC MODEL

3.1 Introduction

In this chapter, we propose a more general scheduled traffic model, called sliding scheduled traffic model. In this model, a demand is represented by a tuple $(s, d, n, \alpha, \beta, \tau)$ that satisfies $\beta - \alpha \geq \tau > 0$ where $s$ and $d$ are the source and destination, $n$ is the number of requested bandwidth units, $\alpha$ and $\beta$ are the starting time and ending time of a time window during which the demand with a holding time of $\tau$ time units resides. In this model, the demand holding time $\tau$ is an interval within a time window $[\alpha, \beta]$. Rather than fixing the starting time and ending time of the demand, a flexibility is introduced in the definition of the interval. As a result, the demand is allowed to slide within a larger time window $[\alpha, \beta]$. This model allows an application to specify a larger time window during which the demand for communication bandwidth is satisfied. Fixing the starting time and ending time of a demand may be too restrictive in practical scenarios. Furthermore, this model gives a service provider more flexibility in provisioning the requested demand and a better opportunity to optimize the network resources since a demand is considered accommodated as long as it is provisioned within the larger time window. Given a demand, the actual starting time of the demand is variable relative to the left boundary $\alpha$ of its associated time window. If the demand starts at $\epsilon$ time units after $\alpha$, the demand is active during $[\alpha + \epsilon, \alpha + \epsilon + \tau]$, as shown in Fig. 3.1.

In this chapter, we derive properties of the sliding scheduled traffic model. Based on the properties, in addition, we develop an efficient algorithm to reduce time conflict among demands by properly place demands within their associated time windows to
minimize their overlapping in the time domain.

The rest of the chapter is organized as follows. We derive properties of the sliding scheduled traffic model in Section 3.2. Section 3.3 presents the demand time conflict reduction algorithm. Section 4.5 summarizes the chapter.

3.2 Properties of Sliding Scheduled Traffic Model

As shown in Fig. 3.1, given a sliding scheduled traffic demand $r$, the actual starting time of the demand is variable relative to the left boundary $\alpha$ of its associated time window. If the demand starts at $\epsilon$ time units after $\alpha$, the demand is active during $[\alpha + \epsilon, \alpha + \epsilon + \tau]$. Let the actual starting time and ending time of demand $r$ be $b^r$ and $e^r$, respectively. Obviously $b^r = \alpha^r + \epsilon^r$ and $e^r = \alpha^r + \epsilon^r + \tau^r$.

Theorem 6: Two sliding scheduled traffic demands $r_p$ and $r_q$ overlap in time if and only if $e^{r_p} > b^{r_q}$ and $e^{r_q} > b^{r_p}$.

**Proof:** We first prove if $e^{r_p} > b^{r_q}$ and $e^{r_q} > b^{r_p}$ then demands $r_p$ and $r_q$ overlap in time. Let $b = \max(b^{r_p}, b^{r_q})$ and $e = \min(e^{r_p}, e^{r_q})$. Since $e^{r_p} > b^{r_p}$ and $e^{r_q} > b^{r_q}$, when $e^{r_p} > b^{r_q}$ and $e^{r_q} > b^{r_p}$, $\min(e^{r_p}, e^{r_q}) > \max(b^{r_p}, b^{r_q})$, i.e., $e > b$. Obviously $[b, e]$ is the common time period of demands $r_p$ and $r_q$. Therefore the two demands overlap in time.

Next we prove if demands $r_p$ and $r_q$ overlap in time then $e^{r_p} > b^{r_q}$ and $e^{r_q} > b^{r_p}$. When $r_p$ and $r_q$ overlap in time, they must have a common time period. Let $[b, e]$ be this time period. Then we have $b^{r_p} \leq b < e \leq e^{r_p}$ for demand $r_p$, and $b^{r_q} \leq b < e \leq e^{r_q}$ for demand $r_q$. Therefore $e^{r_p} > b^{r_q}$ and $e^{r_q} > b^{r_p}$.
This proves the theorem.

We use $T_{r_p,r_q}$ to indicate whether demands $r_p$ and $r_q$ overlap in time ($T_{r_p,r_q}=1$) or not ($T_{r_p,r_q}=0$). In the scheduled traffic model, since the starting time and ending time of each demand are fixed and known in advance, we can easily determine $T_{r_p,r_q}$ for any demand pair $(r_p, r_q)$ and take it as input information for the survivable service provisioning problem, as shown in **ILP3, ILP4, ILP7** and **ILP8**. In the sliding scheduled traffic model, however, it may not be able to determine $T_{r_p,r_q}$ before fixing $e^p$ and $e^q$. One simple case is that once $e^p$ and $e^q$ are fixed, $b^p, e^p, b^q$ and $e^q$ can be determined accordingly. Let $G_{r_p,r_q}^1$ and $G_{r_p,r_q}^2$ be the conditions of $e^p > b^q$ and $e^q > b^p$, respectively. Then we can determine $T_{r_p,r_q}$ by checking conditions $G_{r_p,r_q}^1$ and $G_{r_p,r_q}^2$. If both of them are satisfied, by **Theorem 6**, we conclude that demands $r_p$ and $r_q$ overlap in time (i.e., $T_{r_p,r_q}=1$). Otherwise $T_{r_p,r_q}$ is 0. We determine $T_{r_p,r_q}$ in optimization schemes **ILP9** and **ILP10** presented in Chapter 4.2.

Based on the time windows of a demand set $D$, $\{[\alpha_r, \beta_r] \mid 1 \leq r \leq |D|\}$, we can construct an interval graph $H$ where a vertex of $H$ corresponds to a time window and an edge of $H$ connects two vertices of $H$ if and only if their corresponding time windows intersect (see Figs. 1.5 and 3.2). Now consider the case when the demands

![Fig. 3.2: Interval graph representation of the example demand set in Table 1.1.](image-url)

are placed within their corresponding time windows, we can obtain another interval graph $H'$ by finding the intersection graph of the positioned demand intervals where the vertices of $H'$ is in one-to-one correspondence with the demands and an edge of $H'$ connects two vertices of $H'$ if and only if the corresponding demands intersect
in time. Note that the vertices of $\mathcal{H}$ and $\mathcal{H}'$ are the time windows and associated positioned demands, respectively. Among the edges of $\mathcal{H}$, we distinguish two types: strong edges and weak edges.

**Definition 1:** Consider two nodes $u$ and $v$ of $\mathcal{H}$. Without loss of generality, let $\alpha_u \leq \alpha_v$. An edge $(u, v)$ is strong if and only if $\alpha_u + \tau_u > \beta_v - \tau_v$ holds when $\alpha_u \leq \alpha_v \leq \beta_u \leq \beta_v$, or $\alpha_u + \tau_u > \beta_v - \tau_v$ and $\alpha_v + \tau_v > \beta_u - \tau_u$ hold when $\alpha_u \leq \alpha_v < \beta_v \leq \beta_u$. An edge is weak in all other cases.

From the definition, if vertices of $\mathcal{H}$ corresponding to two time windows are connected by a strong edge, then the corresponding demand intervals intersect in time no matter how they are feasibly placed within their respective windows, i.e., corresponding vertices of $\mathcal{H}'$ are connected with an edge. The set of all strong edges of $\mathcal{H}$ is denoted by $E^S_\mathcal{H}$ while the set of all weak edges is denoted by $E^W_\mathcal{H}$. $E^S_\mathcal{H}$ and $E^W_\mathcal{H}$ are a partition of edges, $E_\mathcal{H}$, of $\mathcal{H}$. As a special case, all edges of $\mathcal{H}$ are strong when only one feasible placement of demand intervals within their time windows exists, that is, $\tau_r = \beta_r - \alpha_r$ for all $r \in D$, in which case the actual starting time and ending time of demands are the same as the starting time and ending time of their respective time windows.

Given two demands connected by a weak edge $(u, v)$ in $\mathcal{H}$, there always exist at least two different ways to suitably place the two demand intervals within their time windows, which results in different interval graphs, of which one graph contains the weak edge $(u, v)$ and the other graph does not. This means that if two demand intervals are suitably placed in their time windows, they may be disjoint in time (i.e., no time overlapping). Therefore, this weak edge may not be present in $\mathcal{H}'$. Hence, an interval graph $\mathcal{H}'$ obtained given a feasible placement of demands is a subgraph of $\mathcal{H}$. In particular, the vertices of $\mathcal{H}$ and $\mathcal{H}'$ correspond to each other, but $E_{\mathcal{H}'} \subseteq E_\mathcal{H}$; furthermore $E^S_{\mathcal{H}} \subseteq E^S_{\mathcal{H}'} \subseteq E_\mathcal{H}$. 

71
Definition 2: A demand \( r = (s, d, n, \alpha, \beta, \tau) \) is tight or its corresponding vertex in \( H \) is tight if \( \beta - \alpha < 2\tau \). Otherwise, a demand or its corresponding vertex in \( H \) is loose.

Lemma 1: No strong edge connects two loose vertices.

**Proof:** Consider two loose vertices \( u, v \) of \( H \), and let \( c_u = \frac{\beta_u - \alpha_u}{2} \) and \( c_v = \frac{\beta_v - \alpha_v}{2} \) be the central coordinate of the corresponding time windows. Assume, without loss of generality, that \( c_u \leq c_v \). Since both vertices \( u, v \) are loose, the demand interval associated with \( u \) placed in the leftmost position (i.e., \( (\alpha_u, \alpha_u + \tau_u) \)) lies completely on the left of \( c_u \), and the demand interval associated with \( v \) placed in its rightmost position (i.e., \( (\beta_v - \tau_v, \beta_v) \)) lies completely on the right of \( c_v \). That is, \( \alpha_u + \tau_u \leq c_u \) and \( c_v \leq \beta_v - \tau_v \). The lemma is proved. \( \square \)

Theorem 7: Let \( v \) be a loose vertex of \( H \), and let \( A(v) \) denote the set of all vertices connected to \( v \) by strong edges. Then all vertices in \( A(v) \) are tight and are pairwise connected by strong edges.

**Proof:** Consider any two vertices \( x, y \in A(v) \). Since edges \( (x, v), (y, v) \) are strong, both \( x \) and \( y \) are tight according to Lemma 1. Therefore, the demand intervals associated with vertex \( x \) and \( y \) intersect the demand interval associated with \( v \). We have
\[
\alpha_v + \tau_v > \beta_x - \tau_x, \quad \alpha_x + \tau_x > \beta_v - \tau_v.
\]
Similarly, we have
\[
\alpha_v + \tau_v > \beta_y - \tau_y, \quad \alpha_y + \tau_y > \beta_v - \tau_v.
\]
Furthermore, since vertex \( v \) is loose, that is \( \beta_v - \tau_v \geq \alpha_v + \tau_v \). Therefore, we have
\[
\alpha_x + \tau_x > \beta_y - \tau_y, \quad \alpha_y + \tau_y > \beta_x - \tau_x.
\]
This proves the theorem. \( \square \)

Note from the theorem that \( \{v\} \cup A(v) \) form a complete subgraph whose edges are all strong. We can also conclude that any complete subgraph of \( H \) with all strong edges has at most one loose vertex.
3.3 Demand Time Conflict Reduction Algorithm

Obviously, reducing overlapping or conflict between demands in time helps temporal resource reuse. In this section, given a set of sliding scheduled traffic demands $D$, we propose a time domain conflict reduction algorithm applied to the set $D$ to find a proper placement of demand intervals in their associated time windows such that the number of demand pairs that overlap (or conflict) in time is minimized. This problem can be solved by first constructing an interval graph $H$ based on the demand time windows. Then all tight and loose vertices as well as all strong and weak edges in the graph $H$ are identified. Note that strong edges reflect time conflicts of demand pairs that cannot be resolved no matter how demand intervals are placed within their time windows. Therefore, strong edges of $H$ are always in $H'$. However, whether a weak edge of $H$ is in $H'$ depends on how demand intervals are placed. Our proposed conflict reduction algorithm works to remove as many weak edges as possible from the graph $H$ to obtain $H'$.

3.3.1 Strong Edge Detection

We first describe how to detect strong edges given two demands in the form $(\alpha_1, \beta_1, \tau_1)$ and $(\alpha_2, \beta_2, \tau_2)$ (for ease of exposition, only relevant timing information is included for a demand). The placements of demand intervals within their corresponding time windows are $\epsilon_1$ and $\epsilon_2$ units of time from their left boundaries, respectively. Feasible placements are subjected to the following constraints:

\[
0 \leq \epsilon_1 \leq \beta_1 - \alpha_1 - \tau_1, \quad \text{and} \\
0 \leq \epsilon_2 \leq \beta_2 - \alpha_2 - \tau_2. \tag{3.1} \leq \epsilon_2 \leq \beta_2 - \alpha_2 - \tau_2. \tag{3.2}
\]

From Definition 1, the demand intervals of the two demands are disjoint in time if

\[
\alpha_1 + \epsilon_1 + \tau_1 \leq \beta_2 - \tau_2, \quad \text{and} \\
\alpha_1 + \epsilon_1 + \tau_1 \leq \beta_2 - \tau_2. \tag{3.3}
\]
\[ \alpha_2 + \epsilon_2 \geq \alpha_1 + \tau_1; \]  
(3.4)

or

\[ \alpha_2 + \epsilon_2 + \tau_2 \leq \beta_1 - \tau_1, \text{ and} \]  
(3.5)

\[ \alpha_1 + \epsilon_1 \geq \alpha_2 + \tau_2. \]  
(3.6)

If Eqs. (3.1), (3.2), (3.3), (3.4), (3.5) and (3.6) result in no solution for \( \epsilon_1 \) and \( \epsilon_2 \), then the vertices associated with these two demands are connected by a strong edge. Otherwise, these two demand intervals may be disjoint in time for some proper feasible placement \( \epsilon_1 \) and \( \epsilon_2 \). The relationship between feasible placements of these two demand intervals (i.e., \( \epsilon_1 \) and \( \epsilon_2 \)) is obviously mandated by

\[ \alpha_1 + \epsilon_1 + \tau_1 \leq \alpha_2 + \epsilon_2, \text{ or} \]  
(3.7)

\[ \alpha_2 + \epsilon_2 + \tau_2 \leq \alpha_1 + \epsilon_1, \]  
(3.8)

as well as Eqs. (3.1) and (3.2).

### 3.3.2 Time Conflict Reduction

Given a set of sliding scheduled traffic demands \( D \), the algorithm (Fig. 3.4) first constructs an interval graph \( H \) based on the time windows \( \{[\alpha_r, \beta_r] \mid 1 \leq r \leq |D| \} \).
//Input: $\mathcal{D} = \{ r = (s_r, d_r, n_r, \alpha_r, \beta_r, \tau_r) \mid 1 \leq r \leq |\mathcal{D}| \}$

**Demand Time Conflict Reduction Algorithm ($\mathcal{D}$)**

1. Construct an interval graph $\mathcal{H}$ based on $\{ [\alpha_r, \beta_r] \mid r \in \mathcal{D} \}$;
2. Identify all the strong edges and tight vertices in $\mathcal{H}$;
3. For each loose vertex in $\mathcal{H}$, find its neighboring vertices directly connected by strong edges in $\mathcal{H}$ (Theorem 1);
4. For each remaining tight vertex, find its neighboring vertices directly connected by strong edges in $\mathcal{H}$;
5. Do
6. Remove as many weak edges as possible from $\mathcal{H}$ using an edge removing policy;
7. Update the feasible placement $\epsilon_r$ of any affected demand interval $r \in \mathcal{D}$ (as a result of removing edges);
8. Until no more weak edges can be removed from $\mathcal{H}$;
9. Use the minimum of all feasible placements $\epsilon_r$ for demand $r \in \mathcal{D}$.

**Fig. 3.4:** Pseudo code for the demand time conflict reduction algorithm.

The algorithm then goes on to identify all strong edges and tight vertices in $\mathcal{H}$. This is achieved by two steps. First, for each loose vertex in $\mathcal{H}$, the algorithm finds its neighboring tight vertices and check if the edge is strong or weak. Second, for each of the remaining tight vertices, the algorithm finds its neighboring tight vertices and check if the edge is strong or weak. These two steps make use of the conditions derived above. Afterwards, the time conflict reduction algorithm uses an edge removing policy to repeatedly remove weak edges in graph $\mathcal{H}$ until no more weak edges can be removed. Finally, for each demand $r$, among all feasible placements that result in reduced time conflict (i.e., $\epsilon_r, 1 \leq r \leq |\mathcal{D}|$ may take various feasible values), the algorithm takes the minimum of $\epsilon_r, 1 \leq r \leq |\mathcal{D}|$ as the final positions for all the demands.

**3.3.3 Weak Edge Removing Policy**

To remove a weak edge between two vertices $u, v$, one needs to determine feasible placements $b_u$ and $b_v$ of the demand intervals associated with $u$ and $v$ that result in no time conflict. This is achieved by solving Eqs. (3.1), (3.2), (3.3), (3.4), (3.5), and (3.6). The edge $(u, v)$ is removed only if the solutions of $b_u$ and $b_v$ are not empty. If edge $(u, v)$ is chosen to be removed, the feasible placements of all vertices that were
connected to \( u \) or \( v \) using weak edges in \( \mathcal{H} \) need to be updated using Eqs. (3.7) and (3.8).

**Most Conflict Resolved First Policy** This policy looks at a vertex, \( v \), that is not in a strong edge connected complete subgraph in \( \mathcal{H} \). It then determines the feasible placements of all vertices (e.g., \( u \)) that are connected to \( v \) using weak edges such that the demand intervals of \( u \) and \( v \) are disjoint in time. Among all the vertex, \( v \), the policy chooses the vertex, \( w \), that results in the most number of weak edges being removed. The policy then removes the removable weak edges connected to \( w \). Therefore, this policy is a greedy approach.

**Most Connected Vertex First Policy** This policy first chooses a vertex, \( v \), that is not in a strong edge connected complete subgraph in \( \mathcal{H} \) and that is connected to the most number of vertices in \( \mathcal{H} \). The policy then determines the feasible placements of all vertices (e.g., \( u \)) that are connected to \( v \) using weak edges such that the demand intervals of \( u \) and \( v \) are disjoint in time. All such edge \((u, v)\) is then removed.

**Least Connected Vertex First Policy** This policy first chooses a vertex, \( v \), that is not in a strong edge connected complete subgraph in \( \mathcal{H} \) and that is connected to the least number of vertices. The policy then determines the feasible placements of all vertices (e.g., \( u \)) that are connected to \( v \) using weak edges such that the demand intervals of \( u \) and \( v \) are disjoint in time. All such edge \((u, v)\) is then removed.

**Earliest Setup Demand First Policy** This policy first chooses a vertex, \( v \), in \( \mathcal{H} \) that corresponds to a demand with earliest setup time, i.e., the earlier a demand starts, the earlier it will be processed. The policy then determines the feasible placements of all vertices (e.g., \( u \)) that are connected to \( v \) using weak edges such that the demand intervals of \( u \) and \( v \) are disjoint in time. All such edge \((u, v)\) is then removed.

Let \( \mathcal{D} = \{ r_1 = (0, 10, 8), r_2 = (3, 28, 5), r_3 = (2, 11, 3), r_4 = (6, 9, 2), r_5 = (12, 18, 3) \} \) (for ease of illustration, only relevant timing information \((\alpha, \beta, \tau)\) is included). The demand time windows \([\alpha, \beta]\) are shown in Fig. 3.3(a). Fig. 3.3(c) is the interval
graph $H$ based on the time windows of the demands where filled vertices are tight and empty vertices are loose; thick edges are strong and thin edges are weak. The greedy edge removing policy is applied. Fig. 3.3(d) shows that all weak edges connected to vertex 2 are removed first. Fig. 3.3(e) shows that all weak edges connected to vertex 3 are removed next. Fig. 3.3(b) shows the final placement of demand intervals. The algorithm properly places the demands in their time windows to reduce demand overlapping in time. There is only one time overlapping between 1 and 4 left.

3.3.4 Numerical and Simulation Results

In this section, we evaluate the performance of our demand time conflict reduction algorithm. We use the NSFNET topology in Fig. 2.2 for performance evaluation and comparison, and use the same simulation settings as that in Chapter 2.2.3.

Figs. 3.5, 3.6 and 3.7 illustrate the percentage of pairwise demand time conflicts resolved under weak, medium, and strong demand time window correlation, respectively. In each scenario, the figures show the percentage for demand sets with demand holding time as 20% - 80% of the time window length (in (a), (b), (c) and (d), respectively), as well as uniformly mixed demand holding time with respect to the time window length (in (e)). From the figures, we observe that the percentage of conflicts resolved decreases as demand holding time increases. The quantity also decreases as time correlation becomes stronger and as the number of demands in a set increases from 10 to 100. We also observe that the earliest setup demand first policy performs the best in almost all cases, especially when the demand time correlation is not very strong.

Fig. 3.8 shows the overall performance of the demand time conflict reduction algorithm with the earliest setup demand first policy under weak, medium, and strong demand time window correlation, respectively. The last figure shows the overall per-
Fig. 3.5: Demand time conflict reduction: Percentage of demand time conflicts resolved for weak time window correlation.

formance of the algorithm given demand sets with uniformly mixed demand holding time with respect to the time window length. From the figure, we observe that well over 50% of time conflicts can be resolved using our proposed algorithm.
3.4 Summary

In this chapter, we have proposed a more general traffic model, called sliding scheduled traffic model. This model gives service providers more flexibility in provisioning the requested demand and a better opportunity to optimize the network resources. We
have derived some useful properties of this traffic model. We have also proposed a demand time conflict reduction algorithm based on these properties. Given a set of sliding scheduled traffic demands $D$, we apply the algorithm to the set $D$ to find a proper placement of demand intervals in their associated time windows such that the
number of demand pairs that overlap (or conflict) in time is minimized. The algorithm constructs an interval graph based on the demand time windows and removes as many weak edges as possible from the graph. Simulation results showed that our proposed demand time conflict reduction algorithm can solve well over 50% of time conflicts.

In the demand time conflict reduction algorithm, in addition, we considered different weak edge removing policies. Among these policies, we observed that the earliest setup demand first policy performs best in almost all scenarios we investigated.

Fig. 3.8: Demand time conflict reduction with earliest setup demand first policy: Percentage of demand time conflicts resolved for weak, medium and strong time window correlation. The last figure shows the result for demand sets with uniformly mixed demand holding time with respect to the time window length.
4. SURVIVABILITY PROVISIONING UNDER SLIDING SCHEDULED TRAFFIC MODEL

4.1 Introduction

In Chapter 3.1, we proposed a general traffic model, called *sliding scheduled traffic model*. This traffic model, similar to the scheduled traffic model yet more general, is able to capture the traffic characteristics of applications that require capacity on a time-limited basis, and gives service providers more flexibility in provisioning the requested demands and a better opportunity to optimize the network resources.

In the work of [56–58], we designed some efficient heuristic algorithms to solve the RWA problem for a single demand or a given demand set, respectively. However, no survivability was considered, i.e., only a working path for a demand was sought. As a matter of fact, there has been no published work that investigates the issue of survivability provisioning in WDM optical networks under the sliding scheduled traffic model.

In this chapter, we study survivable service provisioning with shared protection in wavelength convertible WDM optical mesh networks under the sliding scheduled traffic model. We consider the static version of the problem where a set of sliding scheduled traffic demands, $D$, is given, with the objective of minimizing the total network resources (e.g., number of wavelength-links) used by working paths and protection paths of all the demands in $D$ while 100% restorability is guaranteed against any single failure (in the link failure model) or any single SRLG failure (in the SRLG failure model). Obviously, reducing overlapping or conflict between demands in time helps temporal resource reuse. Therefore, in sliding scheduled traffic model network
resource reuse can be further exploited in time by appropriately placing demands in their associated time windows.

We first develop two joint RWA ILP formulations for dedicated and shared protection schemes, respectively, in survivable WDM optical mesh networks under the sliding scheduled traffic model. We then develop an efficient two-step approach which is based on the demand time conflict reduction algorithm introduced in Chapter 3.3 and the Iterative Survivable Routing algorithm described in Chapter 2.3. In the first step of the approach, we use the demand time conflict reduction algorithm to minimize the time overlapping among a set of demands by properly placing a demand within its associated time window. In the second step, we use the ISR algorithm to find routes and assign wavelengths to the demands whose starting times and ending times have been determined by the demand time conflict reduction algorithm. Finally, we extend the heuristic algorithm design from the single link failure model to the single SRLG failure model.

The rest of the chapter is organized as follows. Section 4.2 presents joint RWA ILP formulations for survivability provisioning under the sliding scheduled traffic model. Section 4.3 describes an efficient 2-step heuristic algorithm based on the demand time conflict reduction algorithm and the ISR algorithm. Section 4.4 extends the heuristic algorithm design from the single link failure model to the single SRLG failure model. Section 4.5 summarizes the chapter.

4.2 Optimization Models

In this section, we develop two joint RWA ILP formulations, **ILP9** and **ILP10**, for dedicated and shared protection schemes in survivable WDM optical mesh networks under the sliding scheduled traffic model, respectively. Similar to protection schemes **DP-S** and **SP-S**, the two protection schemes under the sliding scheduled traffic model are abbreviated as **DP-SS** and **SP-SS**, respectively.
4.2.1 Joint Optimization Schemes

**ILP9 – Joint RWA ILP for DP-SS**

This ILP is developed for joint RWA optimization using dedicated path protection under the sliding scheduled traffic model. In this optimization model, demands are allowed to slide within a larger time window, which gives a better opportunity to optimize the total used network resources since a demand is considered accommodated as long as it is provisioned within the time window.

(1) The following are given as program inputs:

- \( N \): the set of nodes in the network;
- \( L \): the set of links in the network;
- \( K \): the set of wavelengths on each link;
- \( D \): the set of sliding scheduled traffic demands; For each demand \( r = (s_r, d_r, n_r, \alpha_r, \beta_r, \tau_r) \in D \), \( s_r \) and \( d_r \) are the source node and destination node of demand \( r \), respectively, \( n_r \) is the number of requested lightpaths, \( \alpha_r \) and \( \beta_r \) are the starting time and ending time of a time window during which the demand with a holding-time of \( \tau \) time units resides;
- \( M \): a large number (e.g., maximum demand lasting time).

(2) The problem solves the following variables given a set of traffic demands \( D \):

- \( \delta_{i,j}^r \in \{0,1\} \): indicates whether the working path of demand \( r \) traverses link \( (i,j) \) (=1) or not (=0);
- \( \eta_{i,j}^r \in \{0,1\} \): indicates whether the protection path of demand \( r \) traverses link \( (i,j) \) (=1) or not (=0);
- \( A_{i,j}^{r,\lambda} \in \{0,1\} \): indicates whether the working path of demand \( r \) traverses link \( (i,j) \) using wavelength \( \lambda \) (=1) or not (=0);
• $B_{i,j}^{r,\lambda} \in \{0,1\}$: indicates whether the protection path of demand $r$ traverses link $(i,j)$ using wavelength $\lambda$ (=1) or not (=0);

• $\epsilon^r \in \{0,1,2,\ldots\}$: the offset starting time of demand $r$;

• $G_{p,q}^{r,r} \in \{0,1\}$: indicates whether the actual tear-down time of demand $r_p$ is later than the actual setup time of demand $r_q$ (i.e., $e^{r_p} > b^{r_q}$) (=1) or not (=0). To reduce the number of constraints in the ILP, we only consider $G_{p,q}^{r,r} (p < q)$.

Without loss of generality, hereafter we assume that $r_p$ and $r_q$ are ordered in demand set $\mathcal{D}$;

• $G_{p,q}^{r_p,r_q} \in \{0,1\}$: indicates whether the actual tear-down time of demand $r_q$ is later than the actual setup time of demand $r_p$ (i.e., $e^{r_q} > b^{r_p}$) (=1) or not (=0) ($p < q$);

• $T_{i,j}^{r_p, r_q} \in \{0,1\}$: indicates whether demands $r_p$ and $r_q$ overlap in time (=1) or not (=0) ($p < q$);

• $X_{i,j}^{\lambda} \in \{0,1\}$: indicates whether some working paths or protection paths traverse link $(i,j)$ using wavelength $\lambda$ (=1) or not (=0).

Objective

\[
\text{minimize} \left\{ \sum_{\forall (i,j) \in \mathcal{L}} \sum_{\forall \lambda \in \mathcal{K}} X_{i,j}^{\lambda} \right\} \quad (4.1)
\]

Subject to: $(r \in \mathcal{D}, (i,j) \in \mathcal{L}, \lambda \in \mathcal{K}, \text{if not specified otherwise})$

\[
\sum_{\forall o : (s, o) \in \mathcal{L}} \delta_{s, o}^r - \sum_{\forall o : (d, o) \in \mathcal{L}} \delta_{d, o}^r = 0, \forall i : (i, s) \in \mathcal{L} \quad (4.2)
\]

\[
\sum_{\forall i : (i, d) \in \mathcal{L}} \delta_{i, d}^r = 1, \delta_{d, o}^r = 0, \forall o : (d, o) \in \mathcal{L} \quad (4.3)
\]

\[
\sum_{\forall i : (i,j) \in \mathcal{L}} \delta_{i,j}^r - \sum_{\forall j : (j, o) \in \mathcal{L}} \delta_{j, o}^r = 0, \forall j \in \mathcal{N}(j \neq s, d) \quad (4.4)
\]
Eqs. (4.2), (4.3) and (4.4) are to ensure that the flow conservation constraints at source nodes, destination nodes and intermediate nodes are satisfied along working paths, respectively.

\[ \sum_{\forall o: (i, s_r) \in L} \eta_{i,s_r} = 1, \eta_{i,s_r} = 0, \forall i: (i, s_r) \in L \]  \hspace{1cm} (4.5)

\[ \sum_{\forall o: (d_r, o) \in L} \eta_{d_r, o} = 1, \sum_{\forall o: (d_r, o) \in L} \eta_{i,s_r} = 0, \forall i: (i, d_r) \in L \]  \hspace{1cm} (4.6)

\[ \sum_{\forall o: (j, o) \in L} \eta_{j,o} = 0, \forall j \in N \]  \hspace{1cm} (4.7)

Flow conservation constraints along protection paths are satisfied in Eqs. (4.5), (4.6) and (4.7).

\[ \delta_{r,i,j} + \eta_{i,j} \leq 1 \]  \hspace{1cm} (4.8)

Eq. (4.8) guarantees that the working path and protection path of demand \( r \) are link-disjoint.

\[ 0 \leq \epsilon_r \leq \beta_r - \alpha_r - \tau_r \]  \hspace{1cm} (4.9)

Eq. (4.9) guarantees that the actual setup time and tear-down time of demand \( r \) are within the time window during which it resides, i.e., \( b_r = \alpha_r + \epsilon_r \in [\alpha_r, \beta_r] \) and \( e_r = \alpha_r + \epsilon_r + \tau_r \in (\alpha_r, \beta_r] \).

\[ M \times G_{r,p,r,q} \geq (\alpha_r + \epsilon_r + \tau_r) - (\alpha_r + \epsilon_r) + (\alpha_r + \epsilon_r + \tau_r) - (\alpha_r + \epsilon_r), \]  \hspace{1cm} (4.10)

\[ M \times G_{r,p,r,q} \geq (\alpha_r + \epsilon_r + \tau_r) - (\alpha_r + \epsilon_r) + (\alpha_r + \epsilon_r + \tau_r) - (\alpha_r + \epsilon_r), \]  \hspace{1cm} (4.11)

\[ G_{r,p,r,q} + G_{r,q,r,q} \leq T_{r,p,r,q} + 1, \]  \hspace{1cm} (4.12)

Eqs. (4.10) and (4.11) determine whether the conditions \( G_1 \) (i.e., \( e_r - b_r > 0 \)) and \( G_2 \) (i.e., \( e_r - b_r > 0 \)) are satisfied, respectively. Only when both the conditions are
satisfied, \( T^{r_p,r_q} \) will be set to 1, which means demands \( r_p \) and \( r_q \) overlap in time. This is determined in Eq. (4.12).

\[
\sum_{\forall \lambda \in \mathcal{K}} A^{r_p,\lambda}_{i,j} = \delta_{i,j} \times n_r, \quad \sum_{\forall \lambda \in \mathcal{K}} B^{r_q,\lambda}_{i,j} = \eta_{i,j} \times n_r
\] 

(4.13)

Eq. (4.13) guarantees that the requested capacity of demand \( r \) is satisfied.

\[
A^{r_p,\lambda}_{i,j} + A^{r_q,\lambda}_{i,j} + T^{r_p,r_q} \leq 2, \quad A^{r_p,\lambda}_{i,j} + B^{r_q,\lambda}_{i,j} + T^{r_p,r_q} \leq 2,
\]

\[
B^{r_p,\lambda}_{i,j} + A^{r_q,\lambda}_{i,j} + T^{r_p,r_q} \leq 2, \quad B^{r_p,\lambda}_{i,j} + B^{r_q,\lambda}_{i,j} + T^{r_p,r_q} \leq 2,
\]

\( \forall r_p, r_q \in \mathcal{D} \) (4.14)

Eq. (4.14) ensures that if demands \( r_p \) and \( r_q \) overlap in time (i.e., \( T^{r_p,r_q} = 1 \)), then wavelength \( \lambda \) on link \((i, j)\) should not be shared by the working path or protection path of demand \( r_p \) and the working path or protection path of demand \( r_q \). Otherwise, demands \( r_p \) and \( r_q \) are allowed to share the wavelength-link. Therefore we conclude that in this formulation, wavelength-links of the working path of demand \( r \) can be reused by the working path or protection path of other demands that do not overlap in time with \( r \). The same is also true for wavelength-links of the protection path of a demand.

\[
X^{\lambda}_{i,j} \leq \sum_{\forall r \in \mathcal{D}} (A^{r,\lambda}_{i,j} + B^{r,\lambda}_{i,j}), \quad |\mathcal{D}| \times X^{\lambda}_{i,j} \geq \sum_{\forall r \in \mathcal{D}} (A^{r,\lambda}_{i,j} + B^{r,\lambda}_{i,j}).
\] 

(4.15)

Eq. (4.15) determines whether wavelength \( \lambda \) on link \((i, j)\) is used by some working paths or protection paths. The use of a wavelength-link will be counted only once even if it is used by more than one demand at different time instances.

**ILP10 – Joint RWA ILP for SP-SS**

This ILP is developed for joint RWA optimization using shared path protection under the sliding scheduled traffic model. Similar to **SP-S**, this optimization model enables resource optimization in both space and time. In addition, it maximally exploits the time disjointness among demands by sliding them within their time windows to minimize the total network resources used.
This problem solves the following variables in addition to those defined in ILP9:

- $S_{i,j}^{r_p,r_q} \in \{0, 1\}$: indicates whether the working paths of demands $r_p$ and $r_q$ are link-joint with respect to link $(i, j)$ ($=1$) or not ($=0$) ($p < q$);

- $S_{r_p,r_q} \in \{0, 1\}$: indicates whether the working paths of demands $r_p$ and $r_q$ are link-joint ($=1$) or not ($=0$) ($p < q$);

- $T S_{r_p,r_q} \in \{0, 1\}$: indicates whether demands $r_p$ and $r_q$ overlap in time and their working paths are link-joint ($=1$) or not ($=0$) ($p < q$).

**Objective**

$$\text{minimize} \left\{ \sum_{\forall (i,j) \in \mathcal{L}} \sum_{\forall \lambda \in \mathcal{K}} X_{i,j}^\lambda \right\}$$  \hspace{1cm} (4.16)

**Subject to:** ($r \in \mathcal{D}, (i, j) \in \mathcal{L}, \lambda \in \mathcal{K}$, if not specified otherwise)

$$\sum_{\forall o: (s, o) \in \mathcal{L}} \delta_{s,r,o} = 1, \quad \delta_{i,s} = 0, \quad \forall i : (i, s) \in \mathcal{L}$$ \hspace{1cm} (4.17)

$$\sum_{\forall o: (d, o) \in \mathcal{L}} \delta_{d,r,o} = 1, \quad \delta_{d,r} = 0, \quad \forall o : (d, o) \in \mathcal{L}$$ \hspace{1cm} (4.18)

$$\sum_{\forall o: (j, o) \in \mathcal{L}} \delta_{i,j}^r - \sum_{\forall o: (j, o) \in \mathcal{L}} \delta_{j,o}^r = 0, \quad \forall j \in \mathcal{N}(j \neq s, d)$$ \hspace{1cm} (4.19)

Eqs. (4.17), (4.18) and (4.19) are to ensure that the flow conservation constraints at source nodes, destination nodes and intermediate nodes are satisfied along working paths, respectively.

$$\sum_{\forall o: (s, o) \in \mathcal{L}} \eta_{s,r,o} = 1, \quad \eta_{i,s} = 0, \quad \forall i : (i, s) \in \mathcal{L}$$ \hspace{1cm} (4.20)

$$\sum_{\forall o: (d, o) \in \mathcal{L}} \eta_{d,r,o} = 1, \quad \eta_{d,r} = 0, \quad \forall o : (d, o) \in \mathcal{L}$$ \hspace{1cm} (4.21)

$$\sum_{\forall o: (j, o) \in \mathcal{L}} \eta_{i,j} - \sum_{\forall o: (j, o) \in \mathcal{L}} \eta_{j,o} = 0, \quad \forall j \in \mathcal{N}(j \neq s, d)$$ \hspace{1cm} (4.22)
Flow conservation constraints along protection paths are satisfied in Eqs. (4.20), (4.21) and (4.22).

\[ \delta_{r;i,j}^r + \eta_{r;i,j}^r \leq 1 \] (4.23)

Eq. (4.23) guarantees that the working path and protection path of demand \( r \) are link-disjoint.

\[ 0 \leq \epsilon^r \leq \beta^r - \alpha^r - \tau^r \] (4.24)

Eq. (4.24) guarantees that the actual setup time and tear-down time of demand \( r \) are within the time window during which it resides, i.e., \( b^r = \alpha^r + \epsilon^r \in [\alpha^r, \beta^r) \) and \( e^r = \alpha^r + \epsilon^r + \tau^r \in (\alpha^r, \beta^r] \).

\[
M \times G_{1;r;p,q}^r \geq (\alpha^r + \epsilon^r + \tau^r) - (\alpha^q + \epsilon^q), \quad M \times (G_{1;r;p,q}^r - 1) < (\alpha^r + \epsilon^r + \tau^r) - (\alpha^q + \epsilon^q),
\]

(4.25)

\[
M \times G_{2;r;p,q}^r \geq (\alpha^r + \epsilon^r + \tau^r) - (\alpha^p + \epsilon^p), \quad M \times (G_{2;r;p,q}^r - 1) < (\alpha^r + \epsilon^r + \tau^r) - (\alpha^p + \epsilon^p),
\]

(4.26)

\[
G_{1;r;p,q}^r + G_{2;r;p,q}^r \leq T_{r;p,q}^r + 1, \quad G_{1;r;p,q}^r + G_{2;r;p,q}^r \geq 2 \times T_{r;p,q}^r,
\]

\[ \forall r_p, r_q \in \mathcal{D}(p < q) \] (4.27)

Eqs. (4.25) and (4.26) determine whether the conditions \( G_1 \) (i.e., \( e^r - b^q > 0 \)) and \( G_2 \) (i.e., \( e^r - b^q > 0 \)) are satisfied, respectively. Only when both the conditions are satisfied, \( T_{r;p,q}^r \) will be set to 1, which means demands \( r_p \) and \( r_q \) overlap in time. This is determined in Eq. (4.27).

\[
\delta_{i,j}^{r_p} + \delta_{i,j}^{r_q} \leq S_{i,j;r;p,q}^r + 1, \quad \delta_{i,j}^{r_p} + \delta_{i,j}^{r_q} \geq 2 \times S_{i,j;r;p,q}^r, \quad \forall r_p, r_q \in \mathcal{D}(p < q)
\]

(4.28)

Eq. (4.28) determines whether the working paths of demands \( r_p \) and \( r_q \) are link-joint with respect to link \((i, j)\). \( S_{i,j;r;p,q}^r \) is set to 1 only when both \( \delta_{i,j}^{r_p} \) and \( \delta_{i,j}^{r_q} \) take on 1.

\[
S_{i,j;r;p,q}^r \leq \sum_{\forall (i,j) \in \mathcal{L}} S_{i,j;r;p,q}^r, \quad |\mathcal{L}| \times S_{i,j;r;p,q}^r \geq \sum_{\forall (i,j) \in \mathcal{L}} S_{i,j;r;p,q}^r, \quad \forall r_p, r_q \in \mathcal{D}(p < q)
\]

(4.29)
Eq. (4.29) determines whether the working paths of demands \( r_p \) and \( r_q \) are sharing any common link.

\[
T^{r_p,r_q} + S^{r_p,r_q} \leq TS^{r_p,r_q} + 1, \quad T^{r_p,r_q} + S^{r_p,r_q} \geq 2 \times TS^{r_p,r_q}, \quad \forall r_p, r_q \in D(p < q)
\]  

Eq. (4.30) determines whether demands \( r_p \) and \( r_q \) overlap in time and their working paths are link-joint.

\[
\sum_{\forall \lambda \in K} A_{i,j}^{r_p,\lambda} = \delta_{i,j}^r \times n_r, \quad \sum_{\forall \lambda \in K} B_{i,j}^{r_p,\lambda} = \eta_{i,j}^r \times n_r
\]  

Eq. (4.31) guarantees that the requested capacity of demand \( r \) is satisfied.

\[
A_{i,j}^{r_p,\lambda} + A_{i,j}^{r_q,\lambda} + T^{r_p,r_q} \leq 2, \quad A_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda} + T^{r_p,r_q} \leq 2, \quad B_{i,j}^{r_p,\lambda} + A_{i,j}^{r_q,\lambda} + T^{r_p,r_q} \leq 2,
\]  

\[
\forall r_p, r_q \in D(p < q)
\]  

Eq. (4.32) ensures that if demands \( r_p \) and \( r_q \) overlap in time (i.e., \( T^{r_p,r_q} = 1 \)), then wavelength \( \lambda \) on link \((i,j)\) should not be shared by the working path of demand \( r_p \) and the working path or protection of demand \( r_q \). In addition, the protection path of demand \( r_p \) and the working path or protection of demand \( r_q \) cannot share the same wavelength-link either. Note that Eq. (4.32) implies that wavelength-links can be reused by two demands that do not overlap in time, which is similar to \textit{ILP9}.

\[
B_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda} + TS^{r_p,r_q} \leq 2, \quad \forall r_p, r_q \in D(p < q)
\]  

Eq. (4.33) ensures that if demands \( r_p \) and \( r_q \) are link-joint and overlap in time, then wavelength \( \lambda \) on link \((i,j)\) should not be shared by their protection paths. Otherwise, the wavelength-link can be shared. Note that by Eqs. (4.32) and (4.33) this formulation maximally exploits the network resource reuse in both space and time, resulting in the minimization of total network resources used.

\[
X_{i,j}^\lambda \leq \sum_{\forall r \in D} (A_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda}), \quad |D| \times X_{i,j}^\lambda \geq \sum_{\forall r \in D} (A_{i,j}^{r_p,\lambda} + B_{i,j}^{r_q,\lambda}).
\]  

Eq. (4.34) determines whether wavelength \( \lambda \) on link \((i,j)\) is used by some working paths or protection paths. The use of a wavelength-link will be counted only once even if it is used by more than one demand at different time instances.
4.2.2 Numerical and Simulation Results

We present and evaluate the performance of ILP9 and ILP10 in this section. As Section 2.2.3, we use the same example networks in Figs. 2.1 and 2.2 for performance evaluation and comparison. We also use the same simulation settings and investigate the same six scenarios shown in Table 2.1.

Tables 4.1 and 4.2 show the total number of wavelength-links needed to satisfy different traffic demand sets in protection schemes DP-SS and SP-SS, respectively. W, M, S represent weak, medium and strong time correlation, respectively, and L represents the percentage of demand holding time over the time window length. When the demand sets have uniformly mixed demand holding time with respect to the time window length, we say L is random in the tables.

As expected, we observe that the SP-SS scheme uses fewer wavelength-links than the DP-SS scheme in all scenarios. The reason is obvious, that is, shared protection

<table>
<thead>
<tr>
<th>Scenario</th>
<th></th>
<th>ILP9 (DP-SS)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>L=20%</td>
</tr>
<tr>
<td>1</td>
<td>W</td>
<td>12*</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>12*</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>12*</td>
</tr>
<tr>
<td>2</td>
<td>W</td>
<td>25*</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>26*</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>27*</td>
</tr>
<tr>
<td>3</td>
<td>W</td>
<td>38</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>55</td>
</tr>
<tr>
<td>4</td>
<td>W</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>99</td>
</tr>
<tr>
<td>5</td>
<td>W</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>112</td>
</tr>
<tr>
<td>6</td>
<td>W</td>
<td>×</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>×</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>×</td>
</tr>
</tbody>
</table>

Tab. 4.1: Total number of wavelength-links used in protection scheme DP-SS. (* indicates the optimal solutions found and W, M, S represent weak, medium and strong time correlation, respectively.)
performs better than dedicated protection given the same network resources. We also observe that as the demand holding time increases, more wavelength-links are needed to accommodate the same demand set. When \( L \) is 100\%, the sliding scheduled traffic model degenerates to its special case, the scheduled traffic model, in which the demand holding time is equal to the length of its corresponding time window.

From the table, we can see that in the first two cases, the size of example networks is small; the set of traffic demands is small; and the number of wavelengths on each link is not large. Under such scenarios, both ILP9 and ILP10 are able to achieve the optimal solutions within a reasonable amount of time. In other cases with larger networks and large demand sets, however, they cannot achieve the optimal solutions, as shown in Cases 3, 4 and 5 of Tables 4.1 and 4.2. When the problem size becomes larger (e.g., larger demand set), ILP9 and ILP10 are not even able to find feasible solutions within the time limit we set in the simulations, i.e., 4 hours of execution, as shown in Case 6 of the two tables. The only exception is the scenario in which \( L \) is 100\%. Under this scenario, the survivability provisioning problem under the sliding scheduled traffic model is simplified to the problem under the scheduled traffic model investigated in Chapter 2.2. Obviously the simplified problem has much lower computation complexity and thus can be solved within the time limit, as shown in our work in Chapter 2.2.

### 4.3 Heuristic Algorithm

#### 4.3.1 Two-step Heuristic Approach: ISR$^+$

We use a two-step approach to solve the survivable RWA problem under the sliding scheduled traffic model. In the first step, given a set of sliding scheduled traffic demands \( \mathcal{D} \), we minimize the time overlapping between demands in \( \mathcal{D} \) by using the demand time conflict reduction algorithm described in Chapter 3.3. Then we use the Iterative Survivable Routing (ISR) algorithm described in Chapter 2.3 to solve the
### 4.3.2 Numerical and Simulation Results

We investigate the impact of different weak edge removing policies by applying the heuristic algorithm $ISR^+$ to the NSFNET network with various sizes of demand sets. Then, we evaluate the performance of the heuristic algorithm by comparing it against that of $ILP10$ since shared path protection under the sliding scheduled traffic model is employed in the optimization model as well as the proposed heuristic algorithm.

**Impact of Weak Edge Removing Policy**

In Chapter 3.3, we have studied the demand time conflict reduction algorithm as well as four weak edge removing policies. The simulation results showed that the

<table>
<thead>
<tr>
<th>Scenario</th>
<th>$r$</th>
<th>$L=20%$</th>
<th>$L=40%$</th>
<th>$L=60%$</th>
<th>$L=80%$</th>
<th>$L=100%$</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>W</td>
<td>11*</td>
<td>11*</td>
<td>11*</td>
<td>11*</td>
<td>11*</td>
<td>11*</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>11*</td>
<td>11*</td>
<td>11*</td>
<td>12*</td>
<td>11*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>11*</td>
<td>11*</td>
<td>12*</td>
<td>13*</td>
<td>11*</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>W</td>
<td>24*</td>
<td>24*</td>
<td>24*</td>
<td>24*</td>
<td>24*</td>
<td>24*</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>25*</td>
<td>26*</td>
<td>28*</td>
<td>30*</td>
<td>25*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>26*</td>
<td>28*</td>
<td>30*</td>
<td>33*</td>
<td>28*</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>W</td>
<td>37</td>
<td>37</td>
<td>38</td>
<td>40</td>
<td>43</td>
<td>38</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>48</td>
<td>48</td>
<td>49</td>
<td>52</td>
<td>54</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>53</td>
<td>55</td>
<td>58</td>
<td>64</td>
<td>68</td>
<td>56</td>
</tr>
<tr>
<td>4</td>
<td>W</td>
<td>59</td>
<td>62</td>
<td>66</td>
<td>72</td>
<td>78</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>70</td>
<td>73</td>
<td>78</td>
<td>83</td>
<td>91</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>90</td>
<td>94</td>
<td>102</td>
<td>112</td>
<td>125</td>
<td>98</td>
</tr>
<tr>
<td>5</td>
<td>W</td>
<td>76</td>
<td>78</td>
<td>80</td>
<td>85</td>
<td>92</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>95</td>
<td>97</td>
<td>100</td>
<td>105</td>
<td>114</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>105</td>
<td>108</td>
<td>113</td>
<td>120</td>
<td>130</td>
<td>111</td>
</tr>
<tr>
<td>6</td>
<td>W</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>119</td>
<td>×</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>152</td>
<td>×</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>218</td>
<td>×</td>
</tr>
</tbody>
</table>

*Tab. 4.2: Total number of wavelength-links used in protection scheme SP-SS. (∗ indicates the optimal solutions found and W, M, S represent weak, medium and strong time correlation, respectively.)*

survivable routing problem under the sliding scheduled traffic model by processing demands sequentially and iteratively. We call the extended algorithm as $ISR^+$. It processes demands whose starting times and ending times have been determined in the demand time conflict reduction algorithm.
earliest setup demand first policy performs the best in almost all cases investigated. However, resolving more demand time conflicts for a given sliding scheduled demand set may not necessarily result in less total network resources used to accommodate the whole demand set. This motivates us to investigate the impact of different weak edge removing policies. We apply the heuristic algorithm \( IRS^+ \) by using traffic demand sets with various sizes in the NSFNET topology. Figs. 4.1, 4.2 and 4.3 show the total number of wavelength-links used by the heuristic algorithm with weak, medium, and strong demand time correlation, respectively, versus the number of demands in a demand set. The \( IRS^+ \) algorithm processes demands by employing the Most Conflicting Demand First policy, which has been shown to be the best demand ordering policy in general in Chapter 2.3. From the figures we observe that although the performance resulting from the four weak edge removing policies appear to be close, the \( IRS^+ \) algorithm employing the earliest setup demand first policy to reduce the demand time conflict performs the best in most scenarios investigated. More specifically, the ESDF policy performs the best in 80% of cases and, in other cases, it only needs up to 10% more wavelength-links than the best solutions achieved by one of other three policies. We observed similar results by using other three demand ordering policies.

**Impact of Demand Length Percentage**

Figs. 4.4 (a), (b) and (c) show the total number of wavelength-links used by the heuristic algorithm \( IRS^+ \) with weak, medium, and strong demand time correlation, respectively, versus the number of demands in a demand set when the earliest setup demand first weak edge removing policy is used. The last figure shows the results for demand sets with uniformly mixed demand holding time with respect to the time window length. From the figures we observe that, as expected, more wavelength-links are needed as the percentage of the demand holding time with respect to the time window length increases.
Fig. 4.1: Total number of wavelength-links versus number of demands under weak time window correlation.

Heuristic Algorithm vs. ILP

Table 4.3 shows the total number of wavelength-links used to satisfy different traffic demand sets in the heuristic algorithm $ISR^+$. We compare its performance with that of ILP10, as shown in Table 4.2, because shared path protection under
the sliding scheduled traffic model is employed in both $ISR^+$ and the optimization model. From the tables, we observe that in the first two scenarios, the size of example networks is small; the set of traffic demands is small; and the number of wavelengths on each link is not large. In these scenarios, the optimization model $ILP_{10}$ is solved
with the optimal solutions. \( ISR^+ \) does not achieve optimal solutions; but the results are very close to that of ILP10. In Scenarios 3 through 6, when network size and demand set size get larger, ILP10 cannot be solved to obtain the optimal solutions within the time limit, and cannot even be solved with feasible solutions within the
Fig. 4.4: Total number of wavelength-links versus number of demands under weak, medium and strong time window correlation. The last figure shows the result for demand sets with uniformly mixed demand holding time with respect to the time window length.

time limit, as we mentioned in Chapter 4.2.2. In contrast, the heuristic algorithm IRS achieves much better performance in much less time, and is able to obtain good performance in Scenario 6.

4.4 Heuristic Algorithm with SRLG

In this section, we extend our heuristic algorithm design described in Section 4.3 from the single link failure model to the single SRLG failure model so that 100% restorability is guaranteed against any single SRLG failure in a WDM optical network.
4.4.1 Two-step Heuristic Approach: $ISR^+$-SRLG

We use a two-step approach to solve the survivable RWA problem in a WDM mesh network with SRLGs under the sliding scheduled traffic model. In the first step, given a set of sliding scheduled traffic demands $D$, we minimize the time overlapping between demands in $D$ by using the demand time conflict reduction algorithm described in Chapter 3.3. Then we use the Iterative Survivable Routing (ISR) algorithm with SRLG (i.e., ISR-SRLG) described in Section 2.4 to solve the survivable routing problem under the sliding scheduled traffic model by processing demands sequentially and iteratively. We call the extended algorithm as $ISR^+$-SRLG. It processes demands whose starting times and ending times have been determined in the demand time conflict reduction algorithm.

4.4.2 Numerical and Simulation Results

**Heuristic Algorithm Performance: Different SRLG Sizes**
Figs 4.5 through 4.10 show the total number of wavelength-links required to accommodate different traffic demand sets in the $ISR^+$-SRLG algorithm with different SRLG set sizes when the demand time correlation are weak, medium, and strong, respectively. Similar to the results reported in Section 2.4.3, we observe that as the size of SRLGs increases, more wavelength-links are needed to accommodate the same traffic demand set regardless of the percentage of demand holding time over the time window length and the SRLG generation methods (i.e., localized or non-localized).

We also observe that when the size of SRLGs and/or the size of demand set become large enough (e.g., $\gamma = 3$ and $|D| \geq 30$), the traffic demand set cannot be accommodated based on the limited network resources (100 wavelengths per link), i.e., the algorithm is unable to find a working path and a protection path for all the demands in $D$ that satisfy the shared path protection constraints and provide 100% restorability for all demands to protect against a single SRLG failure. This is true whether the SRLG sets are localized or not, or whether the demand time correlation is strong or weak.

*Heuristic Algorithm Performance: Different Types of SRLGs*

Figs 4.11, 4.12 and 4.13 show the total number of wavelength-links required in the network with localized SRLGs or non-localized SRLGs when the demand time correlation are weak, medium, and strong, respectively. In general, the performance resulting from the two SRLG generation methods (i.e., localized or non-localized) appears to be very close in the figures for the cases we studied.

4.5 **Summary**

In this chapter, we have developed optimization models and efficient heuristic algorithms to solve the survivability provisioning problem in WDM optical networks under the sliding scheduled traffic model. We have developed two joint RWA ILP formulations, $\text{ILP9}$ and $\text{ILP10}$, for dedicated and shared protection schemes, $\text{DP-SS}$
Fig. 4.5: Comparison of SRLG size: Total number of wavelength-links versus number of demands under weak time correlation (part 1).

and SP-SS, in survivable WDM optical mesh networks under the sliding scheduled traffic model. The simulation results showed that ILP9 and ILP10 can achieve better performance than their counterparts in the scheduled traffic model (i.e., ILP3 and ILP4) when the percentage of demand holding time over the time window length is
less than 100%. The reason is, in the sliding scheduled traffic model, network resource reuse can be further exploited in time by appropriately placing demands in their associated time windows. We also observed that when the size of example networks is small, the set of traffic demands is small, and the number of wavelengths on each link is not large, both \textit{ILP9} and \textit{ILP10} were able to achieve the optimal solutions within a reasonable amount of time. With larger networks and larger demand sets, however, they could not achieve the optimal solutions, and were not even able to find feasible solutions within the time limit we set in the simulations (4 hours of execution).

We have proposed an efficient two-step approach, called \textit{ISR$^+$}, to solve the survivable service provisioning problem under the sliding scheduled traffic model. In the first step of \textit{ISR$^+$}, we used the demand time conflict reduction algorithm described

\textbf{Fig. 4.6:} Comparison of SRLG size: Total number of wavelength-links versus number of demands under weak time correlation (part 2).
Fig. 4.7: Comparison of SRLG size: Total number of wavelength-links versus number of demands under medium time correlation (part 1).

in Chapter 3.3 to minimize the time overlapping among a set of demands by properly placing a demand within its associated time window. In the second step, we used the Iterative Survivable Routing algorithm described in Chapter 2.3 to find routes and assign wavelengths to the demands whose starting times and ending times have been
Fig. 4.8: Comparison of SRLG size: Total number of wavelength-links versus number of demands under medium time correlation (part 2).

determined by the demand time conflict reduction algorithm. The simulation results showed that fewer wavelength-links were needed as the percentage of the demand holding time over the time window length decreased. This confirms that network resource reuse can be further exploited in time under the sliding scheduled traffic model compared with the scheduled traffic model.

We have also extended the heuristic algorithm design from the single link failure model to the single SRLG failure model. The extended heuristic algorithm $ISR^+$-SRLG was able to solve the survivable routing problem in WDM optical networks in which 100% restorability is guaranteed against any single SRLG failure under the sliding scheduled traffic model. We considered both localized SRLGs and non-localized SRLGs, as well as SRLG sets with different sizes. The simulation results
showed that as the size of SRLG set increased, more wavelength-links were needed to accommodate a same traffic demand set regardless of the percentage of demand holding time over the time window length and the SRLG generation methods (i.e., localized or non-localized). When the size of SRLG sets and the size of demand set
became large enough, the traffic demand set could not be accommodated based on the limited network resources. The results also showed that, in general, the performance in terms of total number of wavelength-links required by the two SRLG generation methods (i.e., localized or non-localized) appeared to be very close. We observed similar results in the scheduled traffic model.

Fig. 4.10: Comparison of SRLG size: Total number of wavelength-links versus number of demands under strong time correlation (part 2).
Fig. 4.11: Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links versus number of demands under weak time window correlation in 2-SRLG NSFNET network.
Fig. 4.12: Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links versus number of demands under medium time window correlation in 2-SRLG NSFNET network.
(a) Demand length 20%
(b) Demand length 40%
(c) Demand length 60%
(d) Demand length 80%
(e) Uniformly mixed demand length

Fig. 4.13: Comparison of localized SRLGs versus non-localized SRLGs: Total number of wavelength-links versus number of demands under strong time window correlation in 2-SRLG NSFNET network.
5. CONCLUSIONS AND FUTURE WORK

We conclude this dissertation by summarizing our contributions and identifying several new directions for future research in the area of survivability provisioning in WDM optical networks.

5.1 Major Contributions

Survivability provisioning has been an important area of research in WDM optical networks. In this dissertation, we focus on survivability provisioning in WDM optical networks under a scheduled traffic model and a sliding scheduled traffic model that we propose. These two models are able to capture the traffic characteristics of applications that require capacity on a time-limited basis, and also give service providers more flexibility in provisioning the requested demands and a better opportunity to optimize the network resources.

In this work, we have developed various optimization models and efficient heuristic algorithms to solve the survivability provisioning problem in WDM optical networks under the scheduled traffic model. We have developed two joint RWA ILP formulations, ILP1 and ILP2, for dedicated and shared path based protection schemes in survivable WDM optical mesh networks under the conventional static traffic model and two joint RWA ILP formulations, ILP3 and ILP4, under the scheduled traffic model. To solve large problems, we divide the joint RWA problem into a routing subproblem and a wavelength assignment subproblem. We have developed ILP5 for the routing subproblem and ILP6, ILP7 and ILP8 for the wavelength assignment subproblem for protection schemes SP, DP-S and SP-S, respectively.
We have also proposed a capacity provision matrix based Iterative Survivable Routing (ISR) algorithm to solve the survivable service provisioning problem under the scheduled traffic model. We first order the scheduled demands based on different demand ordering policies and then process demands sequentially. The algorithm runs iteratively to approximate the optimal solution. The simulation results showed that the ISR algorithm is extremely time efficient while achieving excellent performance in terms of total network resources used.

We have also extended the heuristic algorithm design from the single link failure model to the single SRLG failure model. The extended heuristic algorithm ISR-SRLG is able to solve the survivable routing problem in WDM optical networks in which 100% restorability is guaranteed against any single SRLG failure under the scheduled traffic model. We consider both localized SRLGs and non-localized SRLGs, as well as SRLG sets with different sizes. The simulation results showed that as the size of SRLG set increases, more wavelength-links are needed to accommodate the same traffic demand set. When the size of SRLGs and/or the size of demand set become large enough, the traffic demand set could not be accommodated based on the limited network resources. This is true whether the SRLG sets are localized or not, or whether the demand time correlation is strong or weak. The simulation results also showed that, in general, the performance resulting from the two SRLG generation methods (i.e., localized or non-localized) appears to be very close in the cases we studied.

We have proposed a sliding scheduled traffic model, which is more general than its special case, the scheduled traffic model, and provides us an opportunity to further optimize the network resources. We have derived some useful properties of this traffic model. Based on these properties, we have proposed a demand time conflict reduction algorithm. Given a set of sliding scheduled traffic demands $\mathcal{D}$, we apply the algorithm to the set $\mathcal{D}$ to find a proper placement of demand intervals in their associated time windows such that the number of demand pairs that overlap (or conflict) in time is
minimized. The algorithm constructs an interval graph based on the demand time windows and removes as many weak edges as possible from the graph. Simulation results showed that our proposed demand time conflict reduction algorithm can solve well over 50% of time conflicts. In the demand time conflict reduction algorithm, we considered different weak edge removing policies. Among these policies, we observed that the earliest setup demand first policy results in the best performance in almost all scenarios that we investigated.

The survivability provisioning problem in WDM optical networks under the sliding scheduled traffic model has never been studied in the research community. In this dissertation, we have developed optimization models and efficient heuristic algorithms to solve the problem. We have developed two joint RWA ILP formulations, ILP9 and ILP10, for dedicated and shared protection schemes DP-SS and SP-SS. The simulation results showed that ILP9 and ILP10 can achieve better performance than their counterparts in the scheduled traffic model (i.e., ILP3 and ILP4) when the percentage of demand holding time over the time window length is less than 100%. We have also proposed an efficient two-step approach ISR+. In its first step, we use the demand time conflict reduction algorithm to minimize the time overlapping among a set of demands by properly placing a demand within its associated time window. In the second step, we use the ISR algorithm to find routes and assign wavelengths to the demands whose starting times and ending times have been determined by the demand time conflict reduction algorithm. Finally, we have extended the heuristic algorithm design from the single link failure model (ISR+) to the single SRLG failure model (ISR+-SRLG). Similar to ISR-SRLG, we consider both localized SRLGs and non-localized SRLGs, as well as SRLG sets with different sizes in the simulation of ISR+-SRLG. We observed that as the size of SRLG set increased, more wavelength-links were needed to accommodate a same traffic demand set regardless of the percentage of demand holding time over the time window length and the SRLG generation methods.
We also observed that, in general, the performance in terms of total number of wavelength-links required by the two SRLG generation methods (i.e., localized or non-localized) appeared to be very close. We observed similar results in the scheduled traffic model.

5.2 Future Work

In this dissertation, we addressed issues of survivability provisioning in WDM optical networks under a scheduled traffic model and a sliding scheduled traffic model. We believe that some of the techniques and methodologies developed and used to deal with these issues, in particular, the optimization models, the capacity provision matrix based Iterative Survivable Routing algorithm and the demand time conflict reduction algorithm may well be extended to resolve related issues in WDM optical networks such as survivable traffic grooming and survivable multicast routing. We foresee several research tasks in the area.

5.2.1 Survivable Traffic Grooming under Scheduled Traffic Models

Applications differ considerably in their bandwidth requirements. Many applications require much less bandwidth than a full wavelength can offer. Occupying a full wavelength for a few megabytes of data results in very poor utilization of network resources. On the other hand, wavelength channels operate at the peak electronic speed, making it extremely expensive to electronically process traffic on all wavelengths and fibers going through a switch. Therefore, WDM systems employ optical add/drop multiplexers (OADMs) which allow a wavelength to either be dropped at a node or optically bypass the node’s electronics. Traffic grooming is constructed as a multiplexing mechanism by which low-rate traffic streams can be appropriately aggregated and assigned to wavelength channels with the objective of efficiently utilizing network resources and minimizing the cost of electronic processing in the network.
We have addressed the traffic grooming problem in WDM optical mesh networks under the sliding scheduled traffic model [56]. To the best of our knowledge, traffic grooming under the sliding scheduled traffic model has not been studied in the literature.

Recently survivable traffic grooming has attracted much attention from researchers which provides reliable service at sub-wavelength level. Like the survivability provisioning problem at the full-wavelength level, however, all existing work concentrated on conventional traffic models, especially on the static traffic model and the dynamic random traffic model. We believe that the survivable traffic grooming problem under the scheduled traffic models can potentially be an excellent area for further exploration. In this problem, network resource reuse can be exploited in space by grooming multiple low-rate traffics on the same wavelength while network resource sharing is maximally exploited in time by taking advantage of the properties of the scheduled traffic model and the sliding scheduled traffic model as we have investigated in this dissertation.

5.2.2 Survivable Multicast Routing under Scheduled Traffic Models

In this dissertation, we considered unicast traffic in the survivability provisioning problem. While multicast routing in WDM networks has received considerable attention, multicast routing under the scheduled traffic models has never been studied in the literature. Multicast is the delivery of information to a group of destinations simultaneously using the most efficient strategy to deliver the messages over each link of the network only once, creating copies only when the links to the destinations split. Under the scheduled traffic models, the multicast routing can utilize the network resources such as wavelength-links and nodes with splitting capabilities more efficiently.

Survivable multicast routing under scheduled traffic models is yet another promis-
ing area that deserves attention. The work reported in [69] and [70] considers establishing a multicast session in a mesh network while protecting the session against any single link failure. Protection of multicast sessions under the scheduled traffic models is an interesting and challenging research topic. The objective can be to minimize the total network resources (e.g., number of wavelength-links and number of nodes with splitting capabilities) used by the multicast spanning trees of a given set of demands while 100% restorability is guaranteed against any single failure. In this problem, network resource sharing can be exploited in both space and time among multicast spanning trees rather than single paths considered in unicast traffic, which makes the problem much more complicated.
Publications

Journal Papers


Conference Papers


**Papers Submitted**


[34] F. Ricciato, S. Salsano, a. Belmonte, and M. Listanti. Off-line configuration of 
a MPLS over WDM network under time-varying offered traffic. *Proceedings of 

[35] N. Srinivas and C. S. R. Murthy. Design and dimensioning of a WDM mesh net-
work to groom dynamically varying traffic. *Photonic Network Communications*,

*Proceedings of SPIE Conference on All-Optical Networking 1999: Architecture, 
Control, and Management Issues, Boston, MA*, September 1999.

[37] S. Arakawa and M. Murata. Lightpath management of logical topology with 
incremental traffic changes for reliable IP over WDM networks. *Optical Network 

[38] N. Geary, A. Antonopoulos, E. Drakopoulos, and J. O’Reilly. Analysis Of Opti-


[40] H. Zang. WDM Mesh Networks: Management and Survivability. *Kluwer Aca-


