WDM Network
Planning and Management

Vom Promotionsausschuss der
Technischen Universität Hamburg-Harburg
zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften
genehmigte Dissertation

Von
Phuong Nga Tran
aus Hanoi, Vietnam

2010

Vorsitzender: Prof. Dr. Siegfried M. Rump

1. Gutachter: Prof. Dr. Ulrich Killat

2. Gutachter: Prof. Dr. Volker Turau

Tag der mündlichen Prüfung: 08.11.2010

Acknowledgement

It was a great experience for me to pursue my Ph.D degree at the Institute of Communication Networks, Hamburg University of Technology.

My special thanks go to Prof. Killat, who did not only give me the opportunity to work in this topic but also supervised me during my research. His extremely helpful instructions kept me always in the right direction, so that I could finally reach my goal.

To all colleagues in the Institute of Communication Networks, I am very grateful for the interesting technical discussions as well as a great working environment.

Last but not least, I would like to thank my family, my parents and my brother, and all my friends for encouraging me to pursue this task. Without their constant support, I would not be able to finish this work.

Abstract

The explosive growth of Internet and bandwidth-intensive networking applications, such as video-on-demand, multimedia conferencing, interactive distance learning, online-games etc. requires extensive new research in high bandwidth transport networks, of which optical networks employing WDM technology is a promising candidate. This thesis aims to solve some problems related to network planning and management for WDM networks.

First, WDM networks with wavelength converters were studied. The problems of wavelength converter placement and wavelength assignment for networks with sparse limited-range wavelength converters were addressed.

Afterwards, the problem of designing logical topologies in multi-layer WDM networks was discussed in detail. Because the traffic is subject to change over time, the logical topology needs to be reconfigured to adapt to new traffic. This thesis proposed an algorithm to deal with this problem. The algorithm can be implemented in centralized as well as in distributed manner. The distributed approach can lend itself to a protocol implementation.

Finally, a protection mechanism to provide survivability for IP/MPLS over WDM networks under a single-link failure was introduced. Simulation for two network instances was carried out to evaluate the performance of the proposed mechanism.

Contents

List of Figures

List of Tables

Abbreviations

ATM	Asynchronous transfer Mode
BB	Branch and Bound
BC	Branch and Cut
DHLP	Dynamic Heavily Loaded Lightpath Protection
EGA	Extended Genetic Algorithm
FDM	Frequency Division Multiplexing
FLR	First Longest Lambda Run
FWM	Four-wave Mixing
HIRA	Hop-based Integrated Routing Algorithm
ILP	Integer Linear Programming
IP	Internet Protocol
ISP	Internet Service Provider
LCF	Least Converter First
LP	Linear Programming
LS	Longest Segment
LSP	Label Switched Path
LSR	Label Switched Router
LAN	Local Area Networks
MAN	Metropolitan Area Networks
MILP	Mixed-Integer Linear Programming
MPLS	Multi-Protocol Label Switching
NP	Non-polynomial
NPM	Network Planning and Management
OXC	Optical Cross-Connect
PSO	Partical Swarm Optimization

QoS	Quality of Service
RWA	Routing and Wavelength Assignment
SBS	Stimulated Brillouin Scattering
SDH	Synchronous Digital Hierarchy
SONET	Synchronous Optical Network
SPF	Shortest Path First
SPM	Self-phase Modulation
SRS	Stimulated Raman Scattering
SRLG	Shared Risk Link Group
VPN	Virtual Private Network
VWP	Virtual Wavelength Path
WDM	Wavelength Division Multiplexing
WP	Wavelength Path
WT	Wavelength Tree
WXC	Wavelength Cross-connect
XPM	Cross-phase Modulation

Chapter 1

Introduction

The old model of a single computer carrying out all computational needs of an organization has been replaced by a network of many separated but interconnected computers. From the time the ARPANET [1] was considered to be the current-day high-speed network, networking technologies have developed a long way and have tremendous influence on technology development in all fields. Providing good communication medium, sharing available resources, improving reliability of services and cost-effectiveness are main advantages of networking that a single computer cannot provide.

The rapid development of computer networks, especially Internet in the last 20 years has raised new challenges for research in this field. The explosive growth of Internet and bandwidth-intensive networking applications, such as video-on-demand, multimedia conferencing, interactive distance learning, online-games etc. requires high bandwidth transport networks of which capacities are much beyond those of the current high speed networks such as ATM (Asynchronous Transfer Mode) or SONET/SDH (Synchronous Optical Network/Synchronous Digital Hierarchy). Thus a demand for reliable networks of high capacities at low cost is continuously rising. This can be achieved with the help of optical networks, in which optical fibers with theoretical capacities up to 50 Tetrabits per second (Tb/s) are used as transportation media. Apart from the huge bandwidth, optical fibers have low cost, low bit error rate, low signal attenuation, low signal distortion and low power requirement. In addition, optical fibers are more secured compared to copper cables from tapping and are also immune to interference and crosstalk [MG02].

---

[1] The first american network

Wavelength Division Multiplexing (WDM), a favourite multiplexing technology in optical communication networks, which combines multiple signals, each at different carrier wavelength to increase capacity, has become popular in the 90s. It supports a cost-effective method to provide multiple transmissions in wavelength domain. With WDM technology, one can expand the transport capacity without laying more fibers. The first WDM system, realized for the first time in the laboratory in 1978, could only combine two signals. But modern WDM systems can handle up to 160 signals and thus can expand a basic 10Gbps fiber to a theoretical total capacity of 1.6Tbps [Gof03]. Because of the huge bandwidth that other technologies cannot offer, optical networks employing WDM technology are considered to be the potential candidate for future wide area backbone networks.

To employ WDM networks efficiently, the concept of lightpath was introduced. A lightpath is a wavelength channel connecting source and destination node without requiring any electronic-optical conversion at the intermediate nodes. Traffic is transmitted from source to destination using lightpaths. This is knowns as wavelength routing. Multiple lightpaths can be established at the same time to form a logical topology, which is implemented on top of the WDM physical layer. This forms a two-layer network architecture. This architecture has many advantages, such as enhanced node processing capability and protocol transparency. With the rapid growth of data traffic and the Internet Protocol (IP) playing a dominant role in networking technology, IP over WDM becomes the right choice for future internet networks. In IP over WDM networks, IP packets are groomed and transported via lightpaths, which are routed in the WDM layer. This avoids the middle ATM or SONET/SDH layer, resulting in significant overhead savings [MG02]. However, the drawback of traditional IP networks is slow packet forwarding due to the long packet processing time. Therefore, *Multiprotocol Label Switching* (MPLS) was proposed to speed up IP packet forwarding. This leads to the proposal of IP/MPLS over WDM as a Next Generation Internet core network.

The steady change of technology as well as the non-stopping increase of network application requires continuous research to revise and refine existing concepts. This dissertation addresses planning, operation, and management issues in two-layer WDM networks. Specific focuses are planning and reconfiguration of logical topologies, which are implemented on top of WDM physical layer to carry IP traffic, survivability of multi-layer architecture networks as well as sparse wavelength converter placement and wavelength assignment in WDM networks. In this thesis, the main method to address these problems is to formulate them as optimization problems and then apply appropriate algorithms to solve them.

1.1 Problem definition

For the convenience of discussion in the following part of the thesis, we address the problem by posing the following questions:

- How to formulate the network planning as an optimization problem and solve the problem for large networks?
- How to place the wavelength converters in WDM networks to reduce the blocking probability?
- How to assign wavelength to an optical path in WDM networks with sparse limited-range wavelength converters?
- How to design an efficient logical topology for two-layer WDM networks?
- How to reconfigure the logical topology of two-layer WDM networks under dynamic traffic changes?
- How to guarantee a survivable two-layer WDM network under single link failure?

These questions do not cover all problems in two-layer WDM networks but within the scope of this thesis, they are studied and answered thoroughly.

1.2 Contribution

The first contribution of this thesis is a novel meta-heuristic algorithm, a so-called iterative optimization algorithm, to solve large optimization problems, the optimal results of which cannot be obtained within a reasonable time. Since optimization softwares can easily get the optimal solution for small problems within a short time, we exploit this advantage to develop our algorithm. The algorithm first divides the large problem into small ones which can be solved exactly by optimization tools. These subproblems are solved sequentially to get their optimal results. The results are then integrated to get the final solution for the original problem. In this thesis, this algorithm is employed to solve network planning problems for WDM networks.

A further contribution is the solution for wavelength converter placement problem and a distributed wavelength assignment algorithm for WDM networks with sparse limited-range wavelength converters. Networks with wavelength converters have lower blocking probability than that without wavelength converters, since the wavelength continuity constraint is relaxed at wavelength converting nodes. The fact that one cannot install a wavelength converter at every node due to its high cost has raised the question which nodes in the network should be equipped with wavelength converters. In this thesis, we introduce for the first time an exact mathematical formulation for wavelength converters placement. This work is reported in [TK08a]. Additionally, wavelength assignment problem was intensively studied but mainly for networks without wavelength converters. Very few studies deal with the sparse presence of wavelength converters in the network, especially limited-range converters. Hence, in this thesis, a novel wavelength assignment algorithm is developed, which ensures to find the wavelength path with the lowest number of converters if it exists and more important, it can be implemented in a distributed manner without additional communication overheads.

The main contribution of this thesis lies on the planning, management and survivability for multi-layer (or more specific, two layers in this work) optical networks. First, a new approach to design logical topologies for WDM networks is introduced. This approach was developed based on the fact that several logical topologies can be implemented on the same physical topology at a time. Hence a logical topology must be designed in such a way that network resources are saved for future use whereas the traffic is efficiently carried. This approach is particularly useful when network infrastructures are fixed and not easy to be expanded. This work was reported in [TK08b]. Another contribution is the solution for the logical topology reconfiguration problem under dynamic traffic changes. Logical topologies are designed to carry traffic optimally. However, traffic is subject to change over time. Thus, logical topologies need to be changed accordingly. The significance of this work is that creating a new logical topology and migrating from an old topology to a new one are solved at the same time and multiple-lightpath change is allowed. Although this problem has been intensively studied, this is the first time a distributed algorithm for reconfiguring the logical topology is presented. This distributed algorithm can lend itself to a protocol implementation. This work was published in [TK08c, TK09]. The last contribution in this category is a novel protection mechanism for two-layer WDM networks. In optical networks, a single link failure may cause a huge data loss due to the high capacity of an optical link. Hence a good protection and restoration machanism is required. The challenge of multilayer protection lies on the fact that link-disjoint paths in the upper layer may not be link-disjoint in the lower layer. This work overcame this challenge by exploiting the p-cycle concept to create a protection topology in the logical layer, which can reduce complexity and increase resource efficieny of the protection mechanism.

1.3 Outline

This thesis is structured as follows.

Chapter 2 gives an introduction of optical networks employing WDM technology. It presents the basic concepts on which this work is based. These are WDM technology, wavelength routed networks and issues in wavelength routed networks including routing wavelength assignment problem, wavelength converting network, logical topology design and reconfiguration, and survivability in two-layer WDM networks.

Chapter 3 discusses network optimization in detail and its application to WDM network planning. A novel meta-heuristic algorithm, a so-called iterative optimization, is introduced. An example of WDM network planning using the proposed iterative optimization is then discussed.

Chapter 4 studies the wavelength converter placement and wavelength assignment in a network equipped with sparse limited-range wavelength converters. A mathematical formulation is presented to solve the wavelength converter placement problem for a static traffic pattern. By applying this formulation to many different random traffic patterns, one can find the nodes at which wavelength converters should be placed. Furthermore, a distributed wavelength assignment algorithm for a network with sparse limited-range wavelengh converters is proposed. The performance of this algorithm is then compared to the best existing ones.

Chapter 5 studies the logical topology design problem for two-layer WDM networks. The problem is formulated as a linear optimization problem and solved for small network examples. The iterative optimization algorithm described in Chapter 3 is used to solve the problem for real-sized networks.

Chapter 6 discusses the reconfiguration of the logical topology under dynamic traffic changes. The problem is first solved exactly in a centralized manner, using optimization formulation. Then an approximation approach is introduced to reduce the complexity of the exact approach. Afterwards, Lagrangian relaxation method is applied to decompose the optimization formulation into subproblems, which can be solved in a distributed manner. The distributed algorithm uses link-state protocol to exchange messages among nodes. The performance of these three approaches is investigated and compared.

Chapter 7 studies the protection and restoration problem in multilayer WDM networks. In this chapter, a frame work based on p-cycle to protect IP/MPLS over WDM networks under a single link failure is presented. The performance of the proposed mechanism is then compared with existing protection approaches.

Chapter 8 summarises these works and gives some outlooks for further development.

Chapter 2

State of the Art

This chapter gives an introduction to all related concepts of WDM networks, on which this thesis is based. We first present the general concepts of WDM optical networks including WDM technology and wavelength-routed network architecture. Afterwards, issues in wavelength-routed networks related to this thesis are discussed. This includes routing and wavelength assignment problem, wavelength converting networks and its wavelength converter placement problem, logical topology design and reconfiguration problems in two-layer WDM networks, and protection and restoration for multilayer networks. These issues do not cover all problems in WDM networks but within the scope of this thesis, they are discussed in detail.

2.1 WDM optical networks

2.1.1 Wavelength Division Multiplexing technology

In optical networks, data is transfered to light pulses and transmitted over optical fibers. Optical fibers consist of a very fine cylinder of glass, which is the core of the fiber, and a concentric layer of glass, the so-called cladding, which is protected by a thin plastic jacket. The core has a lower refractive index than the cladding, so that total internal reflection can occur. When a ray of light from the core approaches the cladding surface with an angle larger than the *critical angle* [2] the ray is completely reflected, which makes light travel

---

[2] The critical angle is the angle of incidence above which total internal reflection occurs.

internally along the core (see Fig. 2.1).

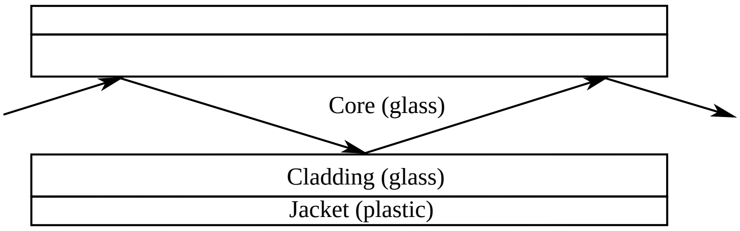


Figure 2.1: Reflection in an optical fiber

Theoretically, the optical fiber can have an extremely high bandwidth, i.e about 25THz, which is 1000 times the total bandwidth of radio on the Earth [CP99]. However, the rate at which an end user can access networks is limited by electronic speed, which is just a few gigabits per second. Hence, it is difficult to exploit all the huge bandwidth of a single optical fiber. Wavelength Division Multiplexing technology has come as a breakthrough, which allows to send many light beams of different wavelengths simultaneously in the core of an optical fiber.

In WDM technology, multiple information signals, each of which corresponds to an end user operating at electronic speed, modulate optical signals at different wavelengths and the resulting signals are then combined and transmitted simultaneously over an optical fiber. This is conceptually similar to Frequency Division Multiplexing (FDM). However, a carrier wave of WDM channel is million times higher than that of an FDM channel infrequency (THz versus MHz). Furthermore, unlike FDM systems, which usually include active devices, WDM systems using diffration grating[3] is completely passive and thus is higly reliable. Illustration of an optical fiber using WDM technology is shown in Fig. 2.2.

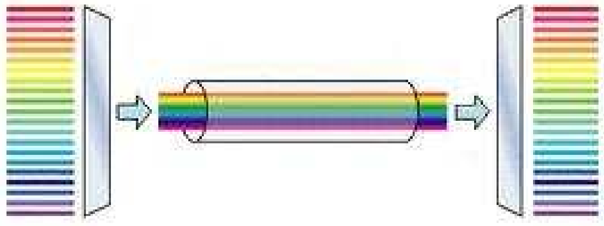


Figure 2.2: Optical fiber using WDM technology

---

[3] A diffraction grating is an optical component with a regular pattern, which splits and diffracts light into several beams travelling in different directions.

The attraction of WDM is that a huge increase in bandwidth can be obtained without necessary investment to deploy additional fibers. WDM has been used to upgrade the capacity of installed transmission systems, by adding several additional wavelengths. Currently, WDM systems using 16 wavelengths at 2.5Gbps and 32 wavelengths at 10Gbps to provide aggregate capacity up to 40Gbps and 320Gbps, respectively, are available. Potential increase in bandwidth per optical fiber thanks to WDM technology is shown in Fig. 2.3. So a transmission system with bandwidth of Tbps is just arround the corner.

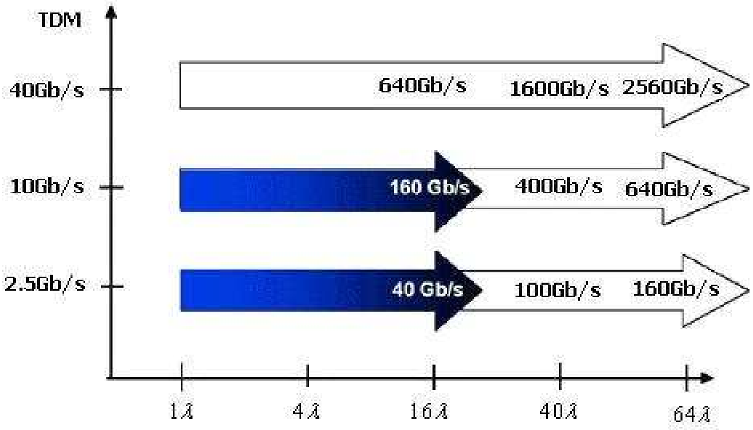


Figure 2.3: Increase in bandwidth per optical fiber

Beside the above-mentioned advantages, WDM systems may face several non-linear effects that limit the performance. These are stimulated Raman scattering (SRS)[4], stimulated Brillouin scattering (SBS)[5], self-phase modulation (SPM), cross-phase modulation (XPM) [6] and four-wave mixing (FWM)[7]. These nonlinearities can be controlled by choosing carefully channel power and channel spacing.

2.1.2 Wavelength routed networks

There are two typical WDM optical network architectures: broadcast-and-select networks and wavelength-routed networks [Muk00]. A broadcast-and select network consists of a passive star coupler connecting nodes in the network, forming a star network. Different nodes transmit data on different wavelengths. The star coupler combines these data and

---

[4] SRS leads to transfer of power from lower-wavelength channels to higher-wavelength channels.

[5] SBS makes the power from the optical signal be scattered back to the transmitter.

[6] SPM and XPM cause phase shifts, which get transformed to signal distortion.

[7] FWM produces new optical frequencies called sidebands causing interference with the data signal.

broadcasts the combined data to all the nodes. A node selects a designed wavelength to receive the desired data. Every node only uses the data that is destined to it and discards the rest. In such networks, the data transmitted by a node is received by all other nodes. Such network architecture has the advantage of simplicity and multicasting capability but in parallel, it has severe limitations. There is no wavelength reuse in these networks and hence a large amount of wavelengths is required. Thus the networks are not scalable beyond the number of supported wavelengths. Furthermore, the transmitted power is splitted among various nodes, so each node just receives a small fraction of the transmitted power. The received signal becomes smaller when the number of nodes increases. Therefore, this network architecture cannot span long distance. The main application for it is high speed local area networks (LAN) and metropolitan area networks (MAN). Wavelength routed networks, different from broadcast-and-select networks, consist of routing nodes interconnected by point-to-point fiber links in an arbitrary topology. They have potential to avoid the problems of broadcast-and-select networks. In this thesis, we only focus on wavelength routed networks. This section will present wavelength routed network architecture in detail.

A wavelength routed network consists of optical cross-connects (OXCs) inter-connected by point-to-point optical fibers in a meshed topology. OXCs are used to switch high-speed optical signals from one fiber to another. These OXCs can be either transparent to signal formats and bit rates, i.e. signals are switched all-optically or be opaque, i.e. signals are first converted from optical to electronic domain and switched electronically. Each node is equipped with a set of transmitters and receivers for sending data into the network and receiving data from the network, respectively.

In a wavelength routed network, data is sent from one node to another node using a wavelength channel, called a lightpath, which is a connection in the optical layer similar to the one in a circuit-switched network. A lightpath does not require any optical-electronic-optical conversion or buffering at any intermediate nodes. The intermediate nodes use their OXCs to route the lightpath in the optical domain. The end nodes of the lightpath access the lightpath using transmitters and receivers that are tuned to the wavelength at which the lightpath operates. Hence, a lightpath is an all-optical communication path, which can be be realized by determining a path in the network between the two nodes and allocating a free wavelength to all links along the path. Generally, there exist two different kinds of optical paths: Wavelength Path (WP) and Virtual Wavelength Path (VWP). In a wavelength path, only one wavelength is used on all links along the path. This is known as *wavelength continuous constraint*. In virtual wavelength path, when wavelength continuity is not required, different wavelengths can be used on different links with the help of wavelength converters.

Two lightpaths cannot use the same wavelength on a fiber. But they can use the same wavelength if they are link-disjoint. This allows simultaneous transmission of messages on the same wavelength over fiber-link-disjoint lightpaths. This property is known as wavelength reuse. Obviously, wavelength reuse makes wavelength routed networks more scalable than broadcast-and-select networks. Another important characteristic of wavelength routed networks is that the transmitted power of a lightpath is not splitted to irrelevant destinations. This enables wavelength routed networks to span long distances from hundreds to thousands of km.

Nowadays, each lightpath has a transmission rate of over Gbps (e.g., OC-48 [2.5Gbps] or OC-192 [10Gbps]). However, the capacity required by traffic streams in IP networks can be significantly lower. In order to achieve the most efficient utilization of network resources as well as to maximize revenue from existing capacity, low speed streams need to be packed onto a high capacity wavelength channel. This is known as *traffic grooming*. Low speed streams that are groomed onto a lightpath may be from different sources and destinations. Thus, a data stream needs to traverse several lightpaths to reach its destination. So a set of lightpaths is needed at the same time to efficiently carry data traffic. This set of lightpaths is known as logical topology because two neigbouring nodes in this topology may not be directly connected in the physical layer. This forms a two-layer network architecture. In fact, multiple logical topologies can co-exist in the same physical network. An example of a two-layer network architecture is illustrated in Fig. 2.4.

One of the wellknown two-layer networks for next-generation carrier networks is IP over WDM. Since IP traffic has exploded in recent years, bandwidth requirements for IP data have reached the limits. So the entire network architecture needs to be reconstructed to cope with exponential growth. In an IP over ATM over SONET over WDM network, 22% bandwidth is used for protocol overhead. Moreover, each layer in this architecture runs at its own speed. So low speed devices cannot fill the whole wavelength bandwidth. Additionaly, many layers do the same function like routing and protection. Therefore a new efficient network architecture is required, which leads to the proposal of IP directly over WDM networks. These networks consist of WDM-aware nodes interconnected by optical fibers. WDM-aware nodes include OXCs as core nodes for routing an optical signal from one fiber to another without performing opto-electronic conversion and edge nodes which

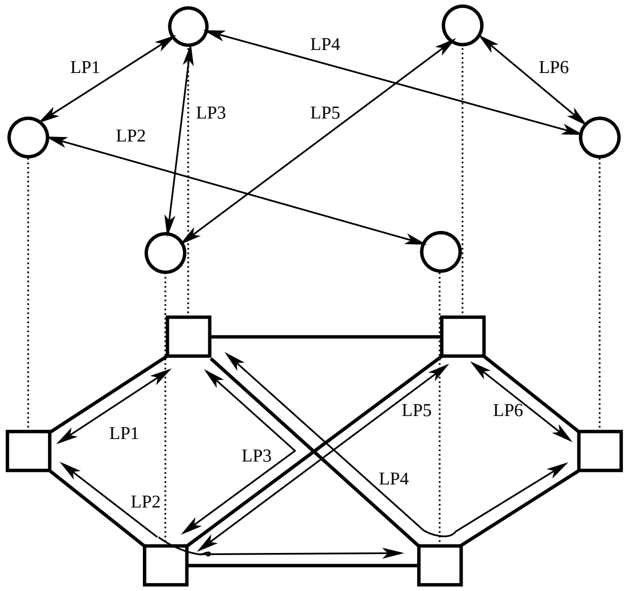


Figure 2.4: Two-layer network architecture

in addition are equipped with transmitters and receivers operating as electronic switches to multiplex and demultiplex IP data packets.

In traditional IP networks, an IP packet is processed by an IP router before being forwarded to the next router. The processing includes examining the IP addresses in the IP packet and determining its next hop by looking up the local routing table. This layer-3 forwarding scheme is rather slow due to the long packet processing time. Hence, the Internet Engineering Task Force (IETF) introduced *Multiprotocol Label Switching* (MPLS) to enable layer-2 forwarding and thus speed up IP packet forwarding [RVC01]. Employing MPLS scheme, IP packets are encapsulated into labeled packets that are forwarded in an MPLS domain along a virtual connection so-called a *Label Switched Path* (LSP). A LSP can be forced to follow a route that is calculated in advance using an explicit routing function. Therefore, MPLS requires processing of short label only, which results in fast forwarding. Furthermore, an LSP can be set up, torn down and rerouted if needed. Bandwidth of an LSP can be modified dynamically according to a specific request. So MPLS can be used in traffic engineering application to optimize resource usage. Currently IP/MPLS over WDM is a potential candidate for Next Generation Internet core networks.

2.2 Issues in Wavelength Routed Networks

2.2.1 Routing and Wavelength Assignment

One of the most important issues in WDM networks is the Routing and Wavelength Assignment (RWA) problem. This is a problem of selecting a route and a wavelength to allocate lightpath requests. Whereas routing is a concern in all networks, wavelength assignment is a unique feature of wavelength-routed networks that differentiate them from conventional networks. Many problems in wavelength routed networks have RWA as a subproblem (e.g. logical topology design, logical topology reconfiguration, and etc). Therefore, it is mandatory to develop a good RWA algorithm to establish lightpaths in an efficient manner.

Typically, lightpath requests can be of two types: static or dynamic. With static traffic, all lightpath requests are known in advance. The aim of static RWA is to allocate as many requests as possible for a given fixed number of wavelengths. Alternatively, one may attempt to allocate lightpath requests so that the network resources such as the number of wavelengths or the number of fibers are minimized. The RWA for static traffic demands is usually performed off-line since connection requests are known beforehand. It can be formulated as an ILP problem [RS95], which is NP-complete[8] [CGK92]. Therefore, it is tractable only for small networks. For larger networks, one usually partitions the problem into two sub-problems: (1) routing and (2) wavelength assignment and each problem can be solved separatedly [BM96].

For the case of dynamic traffic, lightpath requests arrive and depart one by one randomly. The connection is set up for a lightpath when it arrives and after a finite ammount of time, it may be released. The objective of dynamic RWA is usually to minimize the blocking probability or in other words, to maximize the number of connections that are established in the network at any time. Unlike the static RWA problem, which can be solved off-line, any solution to dynamic RWA problem must be computationally simple, as lightpath requests need to be processed online. The RWA for dynamic traffics are generally solved by heuristic algorithms.

Since solving routing and wavelength assignment together is hard, many researchers

---

[8] The time required to solve NP-complete problems using any currently known algorithm increases exponentially as the size of the problem grows.

decouple the problem into two separate subproblems, so-called routing problem and wavelength assignment problem, and solve them sequentially. Usually the routing subproblem is solved first and then wavelength assignment is carried out for the selected route.

Routing problem

In the static case, routing problem can be formulated as an optimization problem and solved by optimization tools. If the problem is large, which cannot be solved optimally by optimization tools within a reasonable time, a heuristic algorithm will be employed. In most of heuristic algorithms, traffic requests are routed sequentially like the case of dynamic traffic using different routing strategies.

In the dynamic case, there are three strategies to solve the routing problem: fixed-routing, fixed-alternative routing and adaptive-routing [CY94, HMM97, LS99, RM98]. In fixed-routing approach, a path (usually the shortest path) for every node pair is calculated offline, using some standard shortest-path algorithms, such as Dijkstra or Bellman-Ford algorithm. Any connection between two nodes is established using this pre-determined route. This approach is very simple but its disadvantage is high blocking probability in dynamic case and large number of wavelengths in static case.

Fixed-alternate routing considers multiple pre-determined routes instead of only one route in fixed-routing. In this approach, every node in the network maintains a table that contains an ordered list of a number of fixed-routes to each destination node. When a lightpath request arrives, the source attemps to establish the connection on each of the routes in the routing table in sequence until a route with a valid wavelength allocation is found. Fixed-alternate routing provides a simple control mechanism of setting up or tearing down lightpaths and offers a significant reduction in blocking probability compared to fixed routing approach [Ram98].

Another routing strategy is adaptive routing. In adaptive routing, a route between two nodes is chosen dynamically based on the network state. Network state is determined by the set of lightpaths currently existing in the network. A common form of adaptive routing is adaptive shortest-cost-path routing. The cost here is calculated based on the network state. There is no common formulation to calculate the cost but depending on prefered criteria of a specific network, a cost function is proposed. Another form of adaptive routing is least-congested-path routing [CY94]. Similar to fixed-alternate routing, in this scheme, a set of routes is pre-selected. Upon the arrival of connection requests, the least-congested-path among the pre-determined routes is chosen. Adaptive routing requires an extensive support from the control and management protocol to continuously update the network state at every node. Hence, its complexity is rather high. The advantage of adaptive routing is that it results in lower blocking probability compared to two beforementioned routing schemes. In summary, fixed-routing is the simplest approach while adaptive routing is the most complex but yealds the best performance. Fixed-alternate routing offers a trade-off between complexity and performance. If the number of alternative routes is sufficiently large, fixed-alternate routing can achieve almost as good performance as adaptive routing. Therefore, fixed-alternative routing is usually prefered.

Wavelength assignment problem

The purpose of the wavelength assignment problem is to assign a wavelength to a lightpath so that it does not share the same wavelength on a given fiber link with another existing lightpath. Wavelength assignment is a unique problem that distinguishes WDM networks from conventional networks. However, this problem exists only if wavelength continuity is required, meaning that no node or just few nodes in the WDM network have the capability of converting a wavelength to another wavelength. This refers to wavelength conversion, which will be discussed in the next section. In the static case, wavelength assignment minimizes the number of wavelengths used under wavelength continuity constraints. This can be formulated as a graph-colouring problem [Muk97]. In the dynamic case, heuristic methods are used to minimize the blocking probability of the network under a given number of wavelengths.

Many heuristic algorithms have been proposed in the literature, such as Random, First-Fit, Least-Used, Most-Used, Least-Loaded, Min-Product, Max-Sum, Relative Capacity Loss, and etc [CGK89, BS97, BK95, JA96, KA98, SB97, ZQ98]. Comparing these wavelength assignment algorithms, [ZJM00] showed that the difference in performance among these various heuristics is not significant. The difference in performance mainly lies in different routing strategies. Among all wavelength assignment algorithms, First-Fit algorithm is the most well-known. In this scheme, all wavelengths are numbered. While searching for an available wavelength, a lower-numbered wavelength is considered before the higher-numbered one. The first available one is then selected. First-Fit is usually prefered in practice because of its small computational overhead and low complexity whereas offering good performance in terms of blocking probability and fairness.

The above-mentioned wavelength assignment algorithms did not take into account the sparse presence of wavelength converters in the network. The use of wavelength converters can reduce the blocking probability but due to their high cost, it is preferred to have just few nodes equipped with wavelength converters. In this case, one has to assign wavelengths so that the number of wavelength conversions on a given path is minimized to save wavelength converters for future use. To solve this problem, [HL06] and [ZY07] proposed similar algorithms called Longest-segment algorithm (LS) and First-longest-lambda-run algorithm (FLR), respectively. However, these algorithms result in the optimal solution only in the case of full range conversion (a wavelength can be converted to any existing wavelength). If the conversion range is limited, the algorithm may either fail to find the lightpath even though it exists or find a non-optimal one, which uses more wavelength converters than needed. The algorithm in [KA96b] can find lightpaths with the least number of converters by constructing an auxiliary graph and apply Dijkstra algorithm to find the shortest path. However, it is complex and requires global information about free wavelengths in the network.

2.2.2 Wavelength Converting Networks

Wavelength continuity can cause the waste of wavelength resouces, resulting in low resource utilization. One possible way to overcome this problem is to use wavelength converters at a routing node. A wavelength converter is an optical device, which can shift a wavelength to another wavelength optically. The capability of wavelength converters can be characterized by the degree of conversion. A converter is called full range converter if it can convert a wavelength to any of other wavelengths. A limited range converter is the one that can convert a wavelength into some neighbour wavelengths. If it can shift a wavelength to any of D other wavelengths, it is said to have conversion degree of D. A WDM network with wavelength converters is called wavelength convertible network. The architecture of an OXC equipped with a wavelength converter is shown in Fig.2.5.

Using wavelength converters, optical links of a lightpath can be assigned different wavelengths. [KA96a] studied the performance in terms of blocking probability of WDM networks with and without wavelength converters using approximate analytical models. The study showed that the performance improvement achieved by convertible networks is mod-

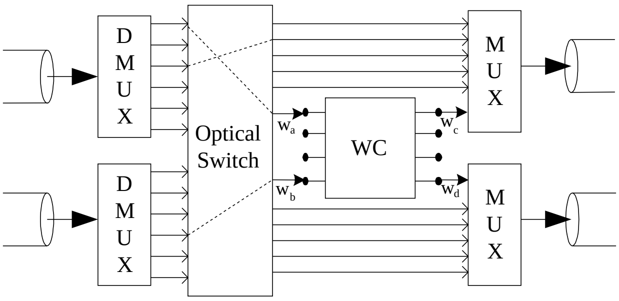


Figure 2.5: Wavelength convertible switch architecture

erate in ring networks but significant in mesh networks. Many other studies have also shown that wavelength converters can improve the performance of large mesh networks, where a path consists of many hops due to the increasing amount of wavelength reuse (10 to 40%, according to Ramaswami et al [RS95]). Wavelength converters relax the wavelength continuity constraint at a node and hence, they help to reduce the bandwidth loss, resulting in better bandwidth utilization. When a network is equipped with full range wavelength converters at every node, it reaches the best achievable performance. However, this network is economically not feasible because wavelength converters are still very expensive. Moreover, the blocking probability does not decrease linearly with the number of converters. According to [SMS96], the rate of performance improvement decreases with the increasing conversion density. Hence sparse wavelength converter placement can achieve almost as good a performance as full wavelength converter placement. This has posed challenges to network designers. It is obviously not neccessary to place converters at all nodes in the network but only at some critical nodes. This raises several issues: Which nodes in the network should be equipped with wavelength converters? How do factors such as traffic demand, network topology, routing scheme affect the allocation of wavelength converters? This is known as wavelength converter placement problem.

The problem of wavelength converter placement has been addressed in many studies. There are two traffic scenarios considered: static [JDH$^+$03, FL07, ZH03, SK04, FCLT05] and dynamic [XL99, VSK99, CL03, GJ03, Xin07, JS05] traffic demands. In all these approaches, researchers typically used heuristic algorithms to place wavelength converters, such as abstracting technique in [ZH03], tabu-search in [SK04], particle swarm optimization (PSO) in [FCLT05] or adaptive traffic-load based in [Xin07]. Some analytical models were also introduced. [XL99, GJ03] and [JS05] use binary linear program to maximize the utilization of wavelength converters, maximize the average of end-to-end success probability and minimize network-wide blocking probability, respectively. However, in these analytical models, they assume that routing is fixed by a single route, so that end-to-end blocking probability can be easily estimated. [CL03] has shown that wavelength converter placement and RWA are closely related and must be solved jointly. A well-designed converter placement mechanism for a specific RWA scheme may not work well for other RWAs [CL03]. Hence, they also proposed some heuristic algorithms to place converters under different RWA algorithms.

2.2.3 Logical topology design

Logical topology in a WDM network consists of a set of lightpaths established among a subset of nodes in the network. The lightpaths are chosen based on the traffic demands among nodes. If two nodes are connected by a lightpath, they are considred to be neighbours in the logical topology and can communicate in one (light) hop. If two nodes are not connected by a lightpath, the communication between them can be achieved by a sequence of lightpaths, resulting in multi-hop communication. Due to the limitations on the number of wavelengths as well as the number of transceivers, it may not be possible to set up a lightpath between all nodes. Therefore, one has to decide which lightpaths may be established so that some specific requirements of the network are satisfied. This leads to the logical topology design problem.

Logical topology design is the problem of selecting and creating (in terms of routing and wavelength assignment) a set of lightpaths under given traffic demands so that these traffic demands are carried efficiently. The objective to create a logical topology is to optimize a certain performance metric, such as message delay, network congestion (maximum load on any lightpath), network throughput and so on. Logical topology design is an important and challenging problem. It requires a good solution to efficiently utilize network resources such as wavelengths and transceivers so as to optimize network performance.

This problem can be divided into three subproblems as follows:

- Determining a logical topology (set of lightpaths) based on a specific objective.
- Doing RWA for the logical topology.
- Routing traffic demands in the logical topology.

These subproblems are not independent and therefore it may lead to a sub-optimal solution if they are solved separatedly. The logical topology design problem has been addressed earlier in several studies [MBRM96,RS96,KS01,BM00,DR00,ZVZO03,LST06,LS05,LP03, KAGS05,JBA06]. Most of conventional research formulated the problem as a mixed-integer linear programming (MILP), which can optimally determine a logical topology subject to the number of transceivers and wavelength constraints, with one of the following objective functions: minimize link utilization [RS96, KS01, ZVZO03], minimize propagation delay [MBRM96], minimize average packet hop distance [BM00] and maximize network throughput [MBRM96,LS05]. Since those optimization problems are NP-hard and thus are not tractable for real-sized networks, some heuristic algorithms were also proposed in the above-mentioned studies. These heuristic algorithms used some standard meta-heuristics such as genetic algorithm, simulated anealling, greedy algorithm or even simply rounded the solutions obtained by solving the relaxed ILP problem [KS01]. Different meta-heuristic algorithms were investigated in [LST06]. The common denominator of those approaches is that they tend to utilize all given resources to design a logical topology so that the network performance is maximized. Recent studies, [LP03,KAGS05,JBA06], addressed the problem while taking resource saving into account. [LP03] formulated the problem as non-linear programming. [KAGS05] proposed a heuristic algorithm, so-called iterative algorithm, to reduce the number of lightpaths. [JBA06] presented an MILP formulation for a survivable logical topology with the objective of minimizing the number of lightpaths or alternatively the number of wavelength-links. The common goal of these approaches is to minimize the cost (in terms of the number of lightpaths or wavelength-links) of the logical topology.

2.2.4 Logical topology reconfiguration

Logical topologies are designed based on the estimated average traffic demands in a specific time frame. The length of this time frame depends on whether the planning is long-term or short-term. The traffic demands among nodes are obviously not constant and subject to change over time. The underlying logical topology, which is optimum for the traffic pattern at the time being created, may not be optimum for current traffic demands. This makes logical topologies outdated and they cannot carry the traffic efficiently. Therefore, reconfiguring logical topologies to be in tune with the changes of traffic demand may help maximizing network performance. Another situation that triggers the reconfiguration is failure restoration. Networks are vulnerable to component failure, which can affect several lightpaths resulting in huge data loss. Therefore, reconfiguration can be used to recover affected lightpahts. In this section, we only focus on the first situation, where reconfiguration is triggered by traffic changes. Reconfiguration due to failure will be discussed in the next section. The flexibility of OXCs to dynamically change the switching patterns of wavelengths from incoming fibers to outgoing fibers aids the reconfiguration process. Reconfigurability is obviously the most valued feature of the WDM networks.

Reconfiguration requires the removal of some lightpaths in the existing logical topology and adding a few lightpaths to form a new logical topology. There are a number of issues related to the logical topology reconfiguration. Migrating one topology to another topology does not only incur control overhead but also service disruption, which is very expensive. Therefore, it is desirable that the new topology is as close to the old one as possible to reduce the lightpath changes. At the same time, the new topology must optimize a certain performance metric to satisfy the network's quality of service.

The reconfiguration of logical topology in two-layer WDM networks has been studied intensively [GB04]. Most of the researchers divide this problem into two sub-problems and solve them separately. The first problem is to find a new optimal logical topology considering several input information including the new traffic demands, current logical topology, and the primary objectives of the design whereas the solution of the second problem finds the best transition from the old topology to the new one with lowest cost. Those two problems attract a fair amount of researchers. There are also different approaches to solve these problems. Some used optimization formulation [GM03, BM00, SL05, AM06, GSM05, BM05] while others used pure heuristic algorithms [RA95, NTM00, ZMT05]. Previous studies usually assumed that the new traffic pattern was known in advance or could be predicted [SM05]. In this case, the algorithm often tries to minimize the traffic loss or minimize the number of lightpaths changes. Some studies addressed the problem for two known logical topologies. They try to move from one to another so that the number of steps is minimzed [BEP$^+$96]. When the future traffic can be predicted, the computation of the new topology can be started well in advance, giving enough time to complete complex computations. Such an approach is similar to off-line topology design.

In practice, the assumption that the future traffic is known, or predictable, is too optimistic. When we are dealing with real data networks, the algorithms and design methods need to be robust enough to deal with unpredictable changes in the traffic. Since the value of the logical topology is highly related to the traffic it is carrying, the reconfiguration method should be flexible to spontaneously reacting to the new conditions. Recently, online adaptation is also considered in [GM03, GSM05, BM05, ZMT05]. Gencata did a step forward by doing reconfiguration without previous knowledge of upcoming traffic [GM03]. The network measures incoming traffic regularly and reacts promptly to the traffic fluctuation. However, this study allows only one lightpath change at a reconfiguration instance. From our point of view, changing only one lightpath is not always efficient, especially for large networks, and consequently, many reconfigurations would be triggered within a short time. [GSM05] allows to add more than one lightpath but is still limited to exactly two or three lightpaths changed at a time.

2.2.5 Survivability in two-layer WDM networks

Because optical networks are prone to component failures while carrying a huge ammount of data, maintaining a high level of service reliability is an important issue. It is mandatory that optical networks have *fault tolerance* capability. Fault tolerance refers to the ability of the network to reconfigure and reestablish the communication upon failure. A network with fault tolerance capability is known as a survivable network. It requires redundant capacity or spare resources to re-allocate traffic once a failure occurs. Survivability for single-layer mesh networks, both conventional and WDM networks, has been investigated thoroughly. Among many protection and restoration mechanisms, *p-cycle* protection, introduced for the first time in [GS98], is known as the most efficient protection mechanism in terms of resource usage and restoration time. The employment of *p-cycle* concept for protection in wavelength-routed networks has been proposed in many lituratures [Mau03, Sch06, Sch05, SSG03, ZY02, ZZB05, ZZM04]. In contrast to single-layer network protection and restoration, multilayer network survivability has been less studied. In this thesis, we focus on survivable multilayer WDM networks, using IP/MPLS over WDM as a network reference. Thus, we will discuss the multilayer protection and restoration in detail in this section.

The survivability issues in multi-layer WDM networks (typically IP/MPLS over WDM networks) have been studied intensively recently [FV00, YADA01, ZM03b, RMZ05, RZG06, MCG$^+$02, SRM02, ZD02]. Recovery can be done either in WDM layer or in IP/MPLS layer. In WDM layer protection, all lightpaths are protected by link-disjoint back-up lightpaths. The attractiveness of this scheme is its fast recovery and low signalling overhead. In fact, in this mechanism, the failure is transparent to IP/MPLS layer. But the main drawback is its poor resource usage and hence high blocking probability. In IP layer protection, each LSPs is protected by a physically link-disjoint back-up LSP. Compared to WDM layer protection, IP layer protection has finer granuality and hence results in better resource usage and blocking performance. However, in IP/MPLS over WDM networks, a lightpath carries many LSPs and an optical link carries many lightpaths, so a single failure can cause a large number of LSPs to be recovered. Therefore, IP layer protection can cause excessive signalling when failure occurs. In addition, restoration time in IP layer protection is significantly higher than that of WDM layer protection.

Recently, hybrid protection schemes, which combines both WDM layer and IP layer protections, have been proposed to find an approriate tradeoff between blocking performance and signalling overhead in IP/MPLS over WDM networks. In [RMZ05], *Dynamic heavily loaded lightpath protection* (DHLP) scheme was proposed. In this proposal, lightpaths carrying a high number of LSPs are protected in WDM layer while LPSs traversing lightly loaded lightpaths are protected in IP layer. This scheme reduces signalling overhead compared to the pure IP-layer protection scheme when failure occurs. However, the network management becomes very complicated. LSP requests come and leave dynamically, so the status of lightpaths also changes dynamically. The network therefore needs to monitor all lightpaths in order to create or release back-up lightpaths for heavily loaded lightpaths when applicable. This causes much signalling overhead to operate the network. In [ZM03b] and [RZG06], a multi-layer protection scheme based on *differentiated QoS* was proposed. The idea of this approach is to categorize traffic into different classes with different priorities. High priority traffic is protected in WDM-layer due to its short recovery time while low priority traffic is protected in IP-layer. This scheme also causes as much signalling overhead to operate the network as in DHLP scheme because a lightpath can be traversed by many LSPs with different priorities.

Chapter 3

Network Planning and Optimization

Network planning and optimization covers all problems addressed in this thesis. This chapter is therefore devoted to introduce some basic understanding related to network planning and optimization. We first review some fundamental concepts, which are intensively used throughout this dissertation. Afterwards, linear programming, a mathematical approach for network planning and optimization is discussed. Finally, this chapter introduces a new meta-heuristic algorithm to solve optimization problems. An illustrative example of the application of this algorithm is also presented.

3.1 Network planning and management

Network Planning and Management (NPM) addresses all activities related to the network development, operation and evolution. NPM activities can be classified by time-scales. Long-term activities are to design or expand the network to meet requirements for a long period of time, usually from months to years. These include for example: physical topology design and capacity expansion. In this thesis, wavelength converter placement is a long-term activity. Medium-term activities include actions that need to be done weekly or monthly, to achieve the convergence towards the established long-term goal. An example of medium-term activities is offline logical topology adjustment in multilayer networks. Short term activities (real-time to hours) incorporate real-time operations such as restoration upon failure or dynamic routing. In this thesis, dynamic reconfiguration of logical topology can be considered as a short-term activity.

Each NPM activity is carried out to obtain a certain *objective* while satisfying some specific *constraints*. These objectives and constraints are usually expressed in the form of network performance or network resources. Some problems may take network performance as objective and network resources as constraints while other problems may consider network resources as objective and network performance as constraints.

Some terminologies, which is often used while studying network planning, are defined as follows.

- Network *topology*: a graph that consists of *nodes* connected by *links*.
- *Traffic demand*: bandwidth or number of connections requested between any two nodes in the network. Traffic demands for the whole network can be described as a traffic matrix, in which each element of the matrix specifies the traffic request between any two nodes.
- *Capacity*: an attribute of a link, which indicates the maximum traffic that can go through that link.

3.2 Optimization approach

3.2.1 Linear programming

A linear program is a mathematical model, in which the aim is to find a set of non-negative values for variables, which maximize or minimize a linear objective function while satisfying a system of linear constraints. A linear program, in which all variables are required to be integers, is called Integer Linear Program (ILP). If just some of variables are integers, the linear program is called Mixed-integer Linear Program (MILP). Linear programming can be applied to various fields of study, including telecommunications. It has been useful in modeling diverse types of problems in planning, routing, scheduling, assignment, and design.

Linear programs are problems that can be expressed in canonical form:

$$\max c^T x \tag{3.1}$$

$$Ax \leq b \tag{3.2}$$

$$x \in R^+ \tag{3.3}$$

where x represents the vector of variables, c and b are vectors of (known) coefficients and A is a (known) matrix of coefficients. $c^T x$ is called the objective function, which can be minimized or maximized. The equation $Ax = b$ is the constraint which determines a convex polytope[9] over which the objective function is optimized.

Pure LP problems can be solved by the wellknown *simplex* algorithm. It has been proved that the optimum value will always be achieved on (at least) one of the vertices of the polytope. Based on this insight, the simplex algorithm goes along edges of the polytope to vertices with higher objective function value. When a local optimum is reached, by convexity it is also the global optimum and the algorithm terminates. It also ends when an unbounded edge is visited, concluding that the problem has no solution. For most of practical applications, the simplex algorithm has been proved to be very efficient.

ILP and MILP problems are solved by a standard algorithm, so-called Branch and Bound (BB). It consists of a systematic enumeration of all candidate solutions, where large subsets of fruitless candidates are discarded *en masse*, by using upper and lower estimated bounds of objective function. The bounds can be found by a *relaxation* technique, which relaxes the integrality property of the variables. The efficiency of BB algorithm depends on the quality of the bounds. Therefore, it is advantageous to find a better bound than the one resulting from the relaxation technique [PM04]. This is the basic idea behind an enhancement of BB algorithm, a so-called Branch and Cut (BC).

To solve optimization problems proposed in this thesis including ILP and MILP, a commercial LP solver called CPLEX [Ilo] is used. For MILP and ILP problems, it exploits the BC algorithm.

3.2.2 Multi-commodity flow problem

Most of planning and management problems in communication networks can be described as multi-commodity flow model and formulated as linear systems. The term multi-commodity comes from the fact that there are multiple demands with different source and sink nodes

---

[9] In elementary geometry, a polytope is a geometric object with flat sides, which exists in any general number of dimensions

that need to be routed in the network simultaneously. The wellknown solution to this problem is based on linear programming [CLRS01]. The problem is NP-complete [EIS76] for integer flows, even only for two commodities.

Given a flow network $G(V, E)$, where edge $(u, v) \in E$ has capacity $c(u, v)$. A source-destination pair $(s, d) \in V^2$ has a traffic demand $D^{s,d}$. The multi-commodity flow problem can be formulated as follows.

There are two ways to formulate the flow conservation: link-path formulation and node-link formulation. Using the link-path formulation, we first find a set of shortest routes for every nodepair in the network. These routes are described by the parameter $\delta^{s,d}_{r,(u,v)}$. $\delta^{s,d}_{r,(u,v)} = 1$ if link (u, v) belongs to route r connecting the source-destination pair (s, d) and $\delta^{s,d}_{r,(u,v)} = 0$ otherwise. We define the decision variable $x^{s,d}_r$ as the fraction of the traffic demand between source and destination node (s, d) traversing route r.

$$\sum_r x^{s,d}_r = D^{s,d} \qquad ,\forall(s,d) \tag{3.4}$$

$$\sum_{s,d} \delta^{s,d}_{r,(u,v)} \cdot x^{s,d}_r \le c(u,v) \qquad ,\forall(u,v) \tag{3.5}$$

Equation 3.4 is the demand satisfaction constraint and Equation 3.5 is the capacity constraint, which ensures that the traffic flowing through a link does not exceed the capacity of the link.

In the node-link formulation, we define the decision variable $f^{s,d}_{u,v}$ as a flow from the source-destination (s, d) passing through the link (u, v).

$$\sum_v f^{s,d}_{u,v} - \sum_v f^{s,d}_{v,u} = \begin{cases} D^{s,d} & s = u \\ -D^{s,d} & d = u \\ 0 & s \neq u, d \neq u \end{cases} \qquad ,\forall(s,d) \tag{3.6}$$

$$\sum_{s,d} f^{s,d}_{u,v} \le c(u,v) \qquad ,\forall(u,v) \tag{3.7}$$

Equation 3.6 is the flow conservation and Equation 3.7 is the capacity constraint.

Normally when the network topology is known in advance, a set of routes for every nodepair can be pre-determined, link-path formulation is prefered due to its lower complexity compared to the node-link formulation.

3.2.3 Meta-heuristic

A meta-heuristic is a heuristic method to solve a general class of computational problems in the hope of obtaining a more efficient or more robust procedure. There are many meta-heuristic algorithms, among which greedy algorithm, genetic algorithm, tabu-search, and simulated annealing are the most popular and widely applied to solve network optimization problems.

To solve optimization problems for real-sized networks, one often has to use a heuristic algorithm because the high complexity of the problem doesn't allow to find the optimal solution within a reasonable time. In this thesis, we develop a new meta-heuristic algorithm, a so-called iterative optimization to solve network optimization problems. This algorithm can be applied to solve an optimization problem in general but in the scope of this thesis, we just investigate its application to network planning and optimization problem.

3.3 Iterative optimization

Network optimization problems are usually NP-complete or NP-hard, and rapidly become intractable when the network size increases. Normally optimization softwares can solve only small problems with a limited number of variables. Hence, we usually have to go for a heuristic algorithm to solve realistic problems, which may not result in the optimal solution but a good one within a reasonable time. Several well-known heuristics were used intensively in network optimization like greedy algorithm, plain local search, genetic algorithm, simulated annealing or the hybridization of two or more approaches like greedy algorithm together with genetic or simulated annealling algorithm [Mul05]. The performance of a heuristic algorithm depends much on the problem itself and the parameters while designing the algorithm.

In this thesis, we introduce a new meta-heuristic algorithm, so-called iterative optimization, to solve large problems, which is applied specially for network planning.

3.3.1 Algorithm description

Optimization tools (LP solvers) cannot solve large problems within a reasonable time but they are very powerful to solve small problems. So our idea is to divide a large problem into many small ones and solve these sub-problems iteratively to get their optimal results, and finally integrate these results to get the solution for the original problem. This algorithm follows the idea of the greedy algorithm of making the locally optimal choice at each stage [CLRS01] with the hope of finding the global optimum. But the main difference of the iterative optimization algorithm and the conventional greedy algorithm is the problem solved at each stage. Let's consider for example a network planning problem with the objective of minimizing the installation cost while satisfying all traffic demands. In the greedy algorithm, at each stage, only one demand is allocated in such a way that the incremental cost is minimized whereas in the iterative optimization, a set of demands, which is a subset of the traffic matrix, is optimally allocated. This may help to obtain a better result than the conventional greedy algorithm.

The iterative optimization algorithm is described in Fig. 3.1.

Diffferent orders of the sub-problems will result in different solutions. To improve the quality of the solution, one may try to solve the sub-problems in different orders and get the best result from all obtained results. How to divide the original problem and how to arrange the sub-problems depend on the problem itself. Like all other meta-heuristic algorithms, one has to design a detailed algorithm for a specific problem.

3.3.2 Illustrative example

We consider the problem of minimizing the number of optical fibers in WDM networks. Given a network topology, its traffic demands, and the number of wavelengths multiplexed in an optical fiber, we have to design a network so that the number of needed fibers is minimized and hence minimize the installation cost.

3.3.2.1 ILP formulation

We first formulate the problem as an ILP problem using the following notations:

- $Z = \{z\}$: denotes the traffic demand of a nodepair.

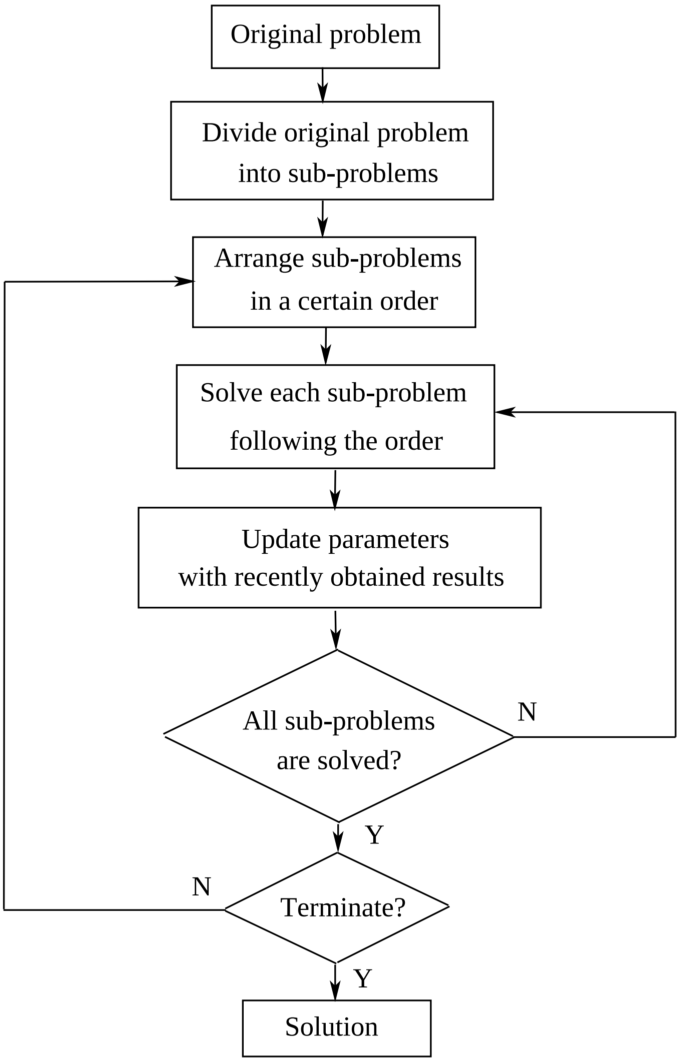


Figure 3.1: Iterative optimization algorithm

- $R = \{r\}$: denotes a set of routes connecting a nodepair.
- $L = \{l\}$: denotes a set of links in the network.
- $\Lambda = \{\lambda\}$: denotes a set of wavelengths multiplexed in a fiber.

Parameters

- d^z: traffic demand of nodepair z
- $\delta^z_{r,l} = 1$ if link l belongs to route r connecting nodepair z. So basically, for every nodepair, a set of R routes is pre-determined in advance.

Variables

- $x^z_{r,\lambda} = 1$ if the wavelength λ is assigned to route r connecting nodepair z. Otherwise, $x^z_{r,\lambda} = 0$.
- e_l: number of fibers installed at link l.

Objective

$$\min \sum_l e_l \tag{3.8}$$

The objective is to minimize the total number of optical links in the network.

Constraints

- Traffic demand constraint:

$$\sum_\lambda^\Lambda \sum_r^R x^z_{r,\lambda} = d^z \qquad \forall z \tag{3.9}$$

- Wavelength capacity constraint:

$$\sum_z^Z \sum_r^R \delta^z_{r,l} \cdot x^z_{r,\lambda} \le e_l \qquad \forall l, \lambda \tag{3.10}$$

Constraint 3.9 ensures that all connection requests are assigned a route and a free wavelength channel. Constrant 3.10 guarantees that at each link, the number of times a wavelength is assigned does not exceed the number of fibers at that link.

3.3.2.2 Heuristic algorithm

The basic idea of the heuristic is to divide the above-mentioned problem into many small sub-problems and to solve these problems consecutively. Because the sub-problems are small (in terms of number of variables), it's possible to solve them within a short time using the LP solver. The question here is how to divide the original problem into sub-problems. There are many different ways to do it and it all depends on the problem itself.

For this problem, we divide the problem based on the traffic demands. As can be seen from the problem, our task is to accommodate a wavelength channel for each demand. Once all the demands are accommodated, the network planning is done. Therefore, for each sub-problem, a subset of demands is considered.

The heuristic algorithm is described in detail as follows.

1. Put all traffic demands in a list.
2. Choose a set of traffic demands in the list to create a sub-problem. This set contains N traffic demands, where N is a parameter of the algorithm. The variation of N may affect the final solution. In this sub-problem, the parameter d^z of the non-selected demands is set to 0. According to Equation 3.9, the related variables $x^z_{r,\lambda}$ will be all 0. This reduces the number of non-zero variables in the problem. Thus, it can be solved easily by the optimization tool.
3. If the sub-problem is the first one to be solved, apply the ILP formulation described above to find the solution for accommodating the set of chosen demands. Once the subproblems are solved, their solutions are stored in the following variables:
   - $f_l = f^*_l + e^*_l$, where f_l is the number of fibers at link l currently, f^*_l is the number of fibers at link l after the previous iteration, and e^*_l is the result obtained from the current iteration.
   - $w_{\lambda,l} = w^*_{\lambda,l} + \sum_{z,r} x^z_{r,\lambda} \cdot \delta^z_{r,l}$, where $w_{\lambda,l}$ is the number of times wavelength λ is assigned at link l currently, $w^*_{\lambda,l}$ is the number of times wavelength λ is assigned at link l after the previous iteration. $x^z_{r,\lambda}$ is the result obtained from the current iteration.

   If the subproblem is not the first one to be solved, we have to slightly change the constraint 3.10 in the ILP model as follows:

   $$\sum_{z}^{Z}\sum_{r}^{R} \delta^z_{r,l} \cdot x^z_{r,\lambda} \leq e_l + f_l - w_{\lambda,l} \qquad ,\forall l, \lambda \tag{3.11}$$

   This is because every sub-problem is solved based on the existing network, which is created by previous iterations. Therefore, the network status needs to be taken into account. Here, $f_l - w_{\lambda,l}$ represents the leftover wavelength resources in the current network.

4. Remove the traffic demands that are just accommodated from the list and return to the step 2 until all demands are routed successfully in the network.

We can shuffle the order of the traffic demands in the list and redo the above steps several times and get the best result out of this.

3.3.2.3 Performance evaluation

To evaluate the performance of the proposed heuristic algorithm, we use the European network with 18 nodes, 39 links as shown in Fig. 3.2 as a network reference for this illustrative example. We assume that there are eight wavelengths multiplexed in a fiber. Five shortest routes are found for each node-pair. CPLEX is used to solve the optimization problem.

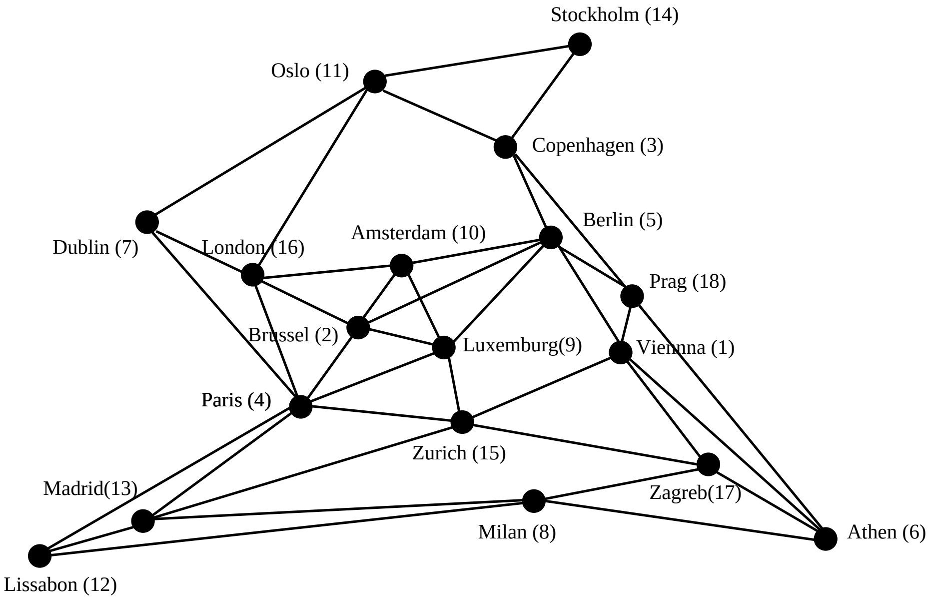


Figure 3.2: European optical network with 18 nodes

We first compare the iterative optimization algorithm with the greedy algorithm, in which only one demand is optimally allocated at each stage. For this purpose, we take the traffic matrix from [Bec01], in which the result of the greedy algorithm was reported, as shown in Table 3.1. In [Bec01], the greedy algorithm tried 1000 different permutations of traffic demands. The statistical results of the greedy algorithm are shown in Fig. 3.3a. We solve the problem using the iterative optimization algorithm for 100 different permutations of traffic demands. The size of the subproblems is seven in this case, meaning that seven traffic demands are considered at each iteration. The results are shown in Fig. 3.3b.

Table 3.1: Traffic matrix, taken from [Bec01]

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	-	1	0	1	5	1	0	2	0	1	0	0	1	1	2	1	1	0
2	-	-	0	3	3	0	0	1	1	2	0	0	1	1	2	2	0	0
3	-	-	-	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0
4	-	-	-	-	6	1	0	3	1	3	1	1	2	1	3	5	0	0
5	-	-	-	-	-	1	0	5	1	4	1	1	2	3	6	4	1	1
6	-	-	-	-	-	-	0	1	0	0	0	0	0	0	1	0	0	0
7	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0
8	-	-	-	-	-	-	-	-	0	1	0	1	2	1	3	1	1	0
9	-	-	-	-	-	-	-	-	-	1	0	0	0	0	1	1	0	0
10	-	-	-	-	-	-	-	-	-	-	0	0	1	1	1	3	0	0
11	-	-	-	-	-	-	-	-	-	-	-	0	0	1	0	1	0	0
12	-	-	-	-	-	-	-	-	-	-	-	-	1	0	1	1	0	0
13	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	1	0	0
14	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	0	0
15	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	0	1
16	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0
17	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0
18	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

From Fig. 3.3, one can see that using the greedy algorithm, the average number of fibers is about 43 whereas the iterative optimization results in 38 fibers on average. More important is that the iterative optimization can get the optimal solution (33 fibers - the solution obtained by CPLEX). Using extended genetic algorithm (EGA) [Bec01], which uses genetic algorithm to shuffle the traffic demands, the best result is 35 fibers, which is still worse than the iterative optimization algorithm. Following the idea of the greedy algorithm of making the locally optimal choice at each stage, the iterative optimization makes use of optimization tools (CPLEX) to improve the greedy algorithm and hence obtains much better results.

We employ the iterative optimization algorithm with different sub-problem sizes, i.e., 7, 9, and 11 demands per iteration. The statistical results are shown in Fig. 3.4. All three approaches can get the optimal results but in average, the larger is the sub-problem size, the better is the solution. The average results for the sub-problem sizes of 7, 9, and 11 are

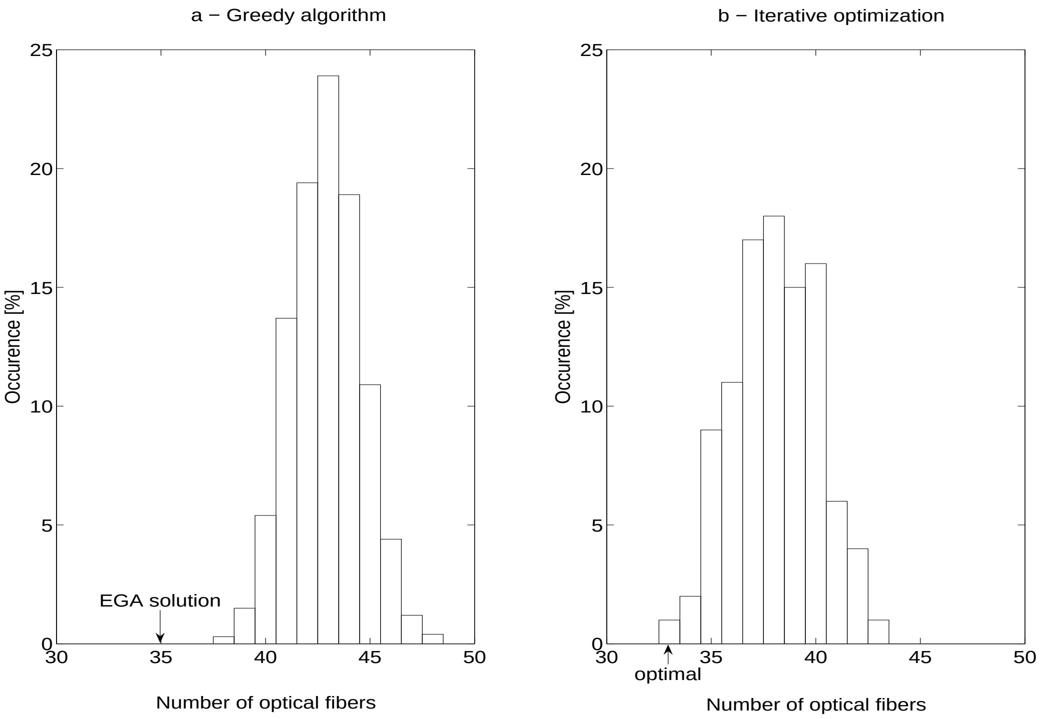


Figure 3.3: Comparison of the iterative optimization and greedy algorithm

38.07, 37.35, and 37.05 fibers, respectively. So increasing the problem size gives a higher chance to obtain a good solution.

For this problem, the time for these three approaches does not differ significantly because the problems are relatively small and hence can be solved within about 15 seconds, including pre-processing time. However, if we increase the problem size further, the solving time can increase sharply. This is because this optimization problem is NP-complete and hence the solving time increases exponentially with the problem size.

The traffic matrix in Table 3.1 has a lot of zero-elements. Hence, the problem is tractable for CPLEX. In this case, it does not make sense to use the iterative optimization. In order to see the advantage of the iterative optimization compared to CPLEX, we solve the WDM network planing problem for a uniform traffic matrix, in which all elements are one. We run the iterative optimization 10 times with different random permutations of traffic demands. The detailed results are reported in Table 3.2.

After one hour, CPLEX got the result of 49 fibers with the gap 13.72%. The total time for the iterative optimization to run 10 times is only 191 seconds and the best result the

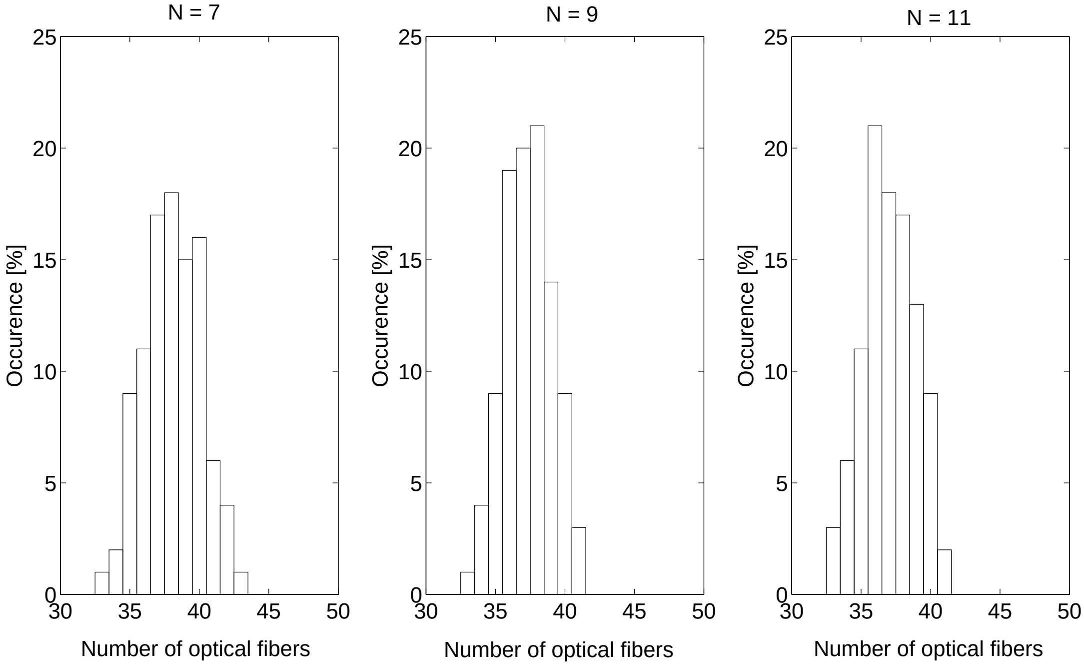


Figure 3.4: Iterative optimization with different sub-problem sizes

iterative optimization can obtain is 44 fibers. The probability that the iterative optimization has a better solution than CPLEX is 50%. Obviously, for large problems, the iterative optimization algorithm is very efficient.

Table 3.2: Comparison of CPLEX and the iterative optimization algorithm

	CPLEX	Iterative optimization (N = 11)									
		1	2	3	4	5	6	7	8	9	10
Number of fibers	49 (gap = 13.72%)	48	50	51	47	46	44	48	49	49	50
Time (sec)	3600	15	13	15	12	14	13	58	17	11	23
		Total solving time = 191									

3.4 Conclusion

This chapter discussed some basic issues related to network planning and optimization, including linear programming, multi-commodity flow problem, and meta-heuristic algorithm. In addition, this chapter introduced a new meta-heuristic algorithm, so-called iterative optimization to solve problems for large networks. The algorithm first divided the original problem into sub-problems, the optimal results of which can be obtained in a reasonable time. These sub-problems were solved sequentially in a specific order. The results were then integrated to get the final solution to the orginal problem. Repetition of the algorithm for different order of sub-problems helps to improve the quality of the solution. An example of WDM network planning employing the proposed algorithm was presented and discussed.

Chapter 4

WDM Networks with Wavelength Converters

This chapter discusses issues in WDM networks deploying wavelength converters. Currently, wavelength converters still remain very expensive. Therefore, it is not cost-effective to equip all nodes with wavelength converters. Moreover, the blocking probability does not decrease linearly with the number of converters. According to [SMS96], the rate of performance improvement decreases with increasing conversion density. Hence sparse wavelength converter placement can achieve almost as good performance as full wavelength converter placement. This is the motivation for studying the wavelength converter placement problem.

Once wavelength converters are placed in the network, how to use the converters is a problem that needs to be solved. Since the number of converters is limited, it is wise to minimize the number of conversions for a lightpath so that converters are saved for future use. Thus, we propose a novel distributed wavelength assignment algorithm to exploit wavelength converters most efficiently.

4.1 Wavelength converter placement

Wavelength converters play an important role in WDM networks because they can improve network performance by relaxing the wavelength continuity constraint. Since only some nodes in the network can be equipped with wavelength converters for economical reasons, choosing the right place for converters is an important issue. This poses a new challenge for network designers. In this section, we introduce a new approach to solve the wavelength converter placement problem. Different from all existing approaches, which use heuristic algorithms, our approach uses ILP, which solves RWA and wavelength converter placement problem at the same time for a static traffic demand. This ILP finds the positions where to place converter so that the total number of necessary converters can be minimized. A large number (e.g. hundreds) of trafffic patterns will be created randomly and the wavelength converter placement problem for each traffic pattern is solved by this optimization model. Based on statistical methods, we then identify the nodes which should be equipped with wavelength converters. To our best knowledge, this is the first optimization model that can solve RWA and wavelength converter placement at the same time. Since wavelength converter placement strongly depends on the RWA scheme [CL03], solving wavelength converter placement together with RWA is necessary.

4.1.1 Problem formulation

We now formulate the problem of minimizing the number of wavelength converters and their placement as an ILP, using the following notation:

- v denotes a physical node.
- l denotes a physical link.
- (l_1, l_2) denotes a pair of consecutive links, which have one common end node.
- z denotes a node-pair.
- λ denotes a wavelength.
- r denotes a route connecting a node-pair.

4.1.1.1 Given parameters

- Physical topology: $F = \{f_l\}$, where f_l indicates the number of optical fibers at link l.

- Lightpath demands: $D = \{d^z\}$, where d^z denotes the number of lightpath requests for the node-pair z.
- P indicates the number of *incoming/outgoing ports* that a wavelength converter can support. This is the number of incoming wavelengths that can be converted to other wavelengths.
- $\delta^z_{r,l}$ denotes the existence of the link l on the route r connecting the node-pair z. $\delta^z_{r,l} = 1$ if l belongs to the route r, and $\delta^z_{r,l} = 0$ otherwise.
- $\theta^z_{r,(l_1,l_2)}$ denotes the existence of consecutive links (l_1, l_2) on the route r connecting the node-pair z. If (l_1, l_2) is on the route r, $\theta^z_{r,(l_1,l_2)} = 1$. Otherwise, $\theta^z_{r,(l_1,l_2)} = 0$.
- $\gamma^v_l = 1$ if node v is one of the two end nodes of link l. Otherwise, $\gamma^v_l = 0$.

4.1.1.2 Decision variables

- Routing for requested lightpaths: y^z_r denotes the number of times route r is used for requested lightpaths of the node-pair z.
- Wavelength assignment: $x^z_{r,l,\lambda}$ denotes the number of times wavelength λ is assigned to the link l, which belongs to the route r connecting the node-pair z. If there is only a single fiber at the link l, $x^z_{r,l,\lambda}$ becomes a binary variable.
- Wavelength conversion: $XC^z_{r,(l_1,l_2),\lambda}$ represents the number of times wavelength λ used on the link l_1 is converted at the common end-node of two consecutive links (l_1, l_2). Similar to $x^z_{r,l,\lambda}$, $XC^z_{r,(l_1,l_2),\lambda}$ becomes a binary variable if there is only a single fiber at the link l.
- Wavelength converter: C_v denotes the number of converters installed at node v.

4.1.1.3 Constraints

- Lightpath routing and wavelength assignment constraints:

$$\sum_r y^z_r = d^z \qquad , \forall z \tag{4.1}$$

$$y_r^z \cdot \delta_{r,l}^z = \sum_{\lambda} x_{r,l,\lambda}^z \qquad ,\forall l, z, r \tag{4.2}$$

$$\sum_z \sum_r x_{r,l,\lambda}^z \leq f_l \qquad ,\forall l, \lambda \tag{4.3}$$

Equation (4.1) represents the routing of requested lightpaths of the node-pair z. Equation (4.2) represents the wavelength assignment on a link. The left-hand side of the equation is the number of lightpaths of the node pair z going through the link l. The right-hand side is the number of wavelengths on the link l used for lightpaths of the node-pair z going through the link l. So the equation ensures that, to all lightpaths, free wavelengths are assigned. Equation (4.3) ensures that the number of times a wavelength is used on a link is lower than or equal to the number of fibers on that link so that a wavelength is used only once in each fiber.

– Wavelength conversion constraints:

$$XC_{r,(l_1,l_2),\lambda}^z = \begin{cases} \theta_{r,(l_1,l_2)}^z (x_{r,l_1,\lambda}^z - x_{r,l_2,\lambda}^z) & \textit{if } x_{r,l_1,\lambda}^z > x_{r,l_2,\lambda}^z \\ 0 & \textit{otherwise} \end{cases} \qquad ,\forall z, r, (l_1, l_2), \lambda \tag{4.4}$$

$$\sum_z \sum_r \sum_{(l_1,l_2)} \sum_{\lambda} XC_{r,(l_1,l_2),\lambda}^z \cdot \gamma_{l_1}^v \gamma_{l_2}^v \leq P{\cdot}C_v \qquad ,\forall v \tag{4.5}$$

Equation (4.4) represents the wavelength conversion between a pair of consecutive links (l_1, l_2). $(x_{r,l_1,\lambda}^z > x_{r,l_2,\lambda}^z)$ means that wavelength λ is used on the link l_1 but not on the link l_2. Hence, there must be a conversion of the wavelength λ at the common end node of these consecutive links. Constraint (4.5) ensures that the total number of conversions at a node does not exceed the conversion capacity provided at that node.

4.1.1.4 Objective

$$\min \sum_v C_v \tag{4.6}$$

Obviously, our objective is to minimize the number of necessary converters in the network. However, one converter can offer several conversions. So it is also desired to minimize the number of conversions. We therefore come up with the following objective function.

$$\min \sum_{v} C_v + \alpha \sum_{z} \sum_{r} \sum_{(l_1,l_2)} \sum_{\lambda} XC^z_{r,(l_1,l_2),\lambda} \tag{4.7}$$

where α is small enough so that the added term does not affect the main objective function in (4.6).

4.1.1.5 Linearization of the non-linear constraint

As can be seen, the constraint (4.4) is not a linear function. However, this constraint can be linearized as follows:

$$XC^z_{r,(l_1,l_2),\lambda} \geq 0 \qquad ,\forall z, r, (l_1, l_2), \lambda \tag{4.8}$$

$$XC^z_{r,(l_1,l_2),\lambda} \geq \theta^z_{r,(l_1,l_2)}(x^z_{r,l_1,\lambda} - x^z_{r,l_2,\lambda}) \qquad ,\forall z, r, (l_1, l_2), \lambda \tag{4.9}$$

A following illustrative example will explain the linearization. Let us consider the equation (4.4). We assume that wavelength λ_1 on link l_1 needs to be converted to wavelength λ_2 on link l_2 for the lightpath connecting the node-pair z, which is routed through the route r. This results in:

$$\begin{array}{lcl} x^z_{r,l_1,\lambda_1} = 1 & \multirow{2}{*}{and} & x^z_{r,l_1,\lambda_2} = 0 \\ x^z_{r,l_2,\lambda_2} = 1 & & x^z_{r,l_2,\lambda_1} = 0 \end{array}$$

Based on the equation (4.4), we then have:

$$XC^z_{r,(l_1,l_2),\lambda_1} = XC^z_{r,(l_2,l_1),\lambda_2} = 1$$
$$XC^z_{r,(l_2,l_1),\lambda_1} = XC^z_{r,(l_1,l_2),\lambda_2} = 0$$

Based on (4.8) and (4.9) in the linearized formulation, we have:

$$\begin{array}{lcl} XC^z_{r,(l_1,l_2),\lambda_1} \geq 1 & \text{and} & XC^z_{r,(l_2,l_1),\lambda_2} \geq 1 \\ XC^z_{r,(l_1,l_2),\lambda_2} \geq 0 & & XC^z_{r,(l_2,l_1),\lambda_1} \geq 0 \end{array}$$

The problem here is that these four variables ($XC^z_{r,(l_1,l_2),\lambda_1}$, $XC^z_{r,(l_1,l_2),\lambda_2}$, $XC^z_{r,(l_2,l_1),\lambda_1}$, and $XC^z_{r,(l_2,l_1),\lambda_2}$) can take many values, because by definition $XC^z_{r,(l_1,l_2),\lambda}$ is not necessary binary if multi-fibers are used. However, in the objective function, we also minimize the summation of all conversions. So once $XC^z_{r,(l_1,l_2),\lambda}$ can take many different values, the lowest one will be chosen for the optimal solution, meaning that for the optimal solution, we have $XC^z_{r,(l_1,l_2),\lambda_1} = XC^z_{r,(l_2,l_1),\lambda_2} = 1$ and $XC^z_{r,(l_1,l_2),\lambda_2} = XC^z_{r,(l_2,l_1),\lambda_1} = 0$. In case no conversion is needed, they will take the value of zero.

Now we have a complete integer linear programming formulation for the problem of wavelength converter placement. Obviously, the solution space of the linearized problem is not exactly the same as the one of the original problem but these two problems have the same optimal solution.

Note: both variables $XC^z_{r,(l_1,l_2),\lambda_1}$ *and* $XC^z_{r,(l_2,l_1),\lambda_2}$ *represent only one conversion. So in the objective function, while summing up all XC, a conversion is actually counted twice. It is not necessary to change the objective function due to the presence of the coefficient* α*. But when calculating the number of conversions, the summation must be divided by 2.*

4.1.2 Illustrative numerical examples

We carry out some experiments with three different network topologies: 9-node regular network, NFSNET (14 nodes) and a German network (17 nodes) as shown in Fig. 4.1, assuming that there are four wavelengths multiplexed in an optical fiber and three shortest-paths for every node-pair.

4.1.2.1 Nodes with high probability of converter placement

For a given traffic pattern, by applying the optimization model introduced in Section 4.1.1, we can find exactly the minimum number of necessary wavelength converters, their placements as well as RWA of all lightpaths on the network. The results show that, for a given traffic pattern, we usually need only one, two or maximum three conversions. This confirms that sparse wavelength converters can achieve the same performance as full conversion.

However, the result will differ if traffic changes. So we prefer to know the nodes for which being equipped with wavelength converters pays off. To see this, we will use a

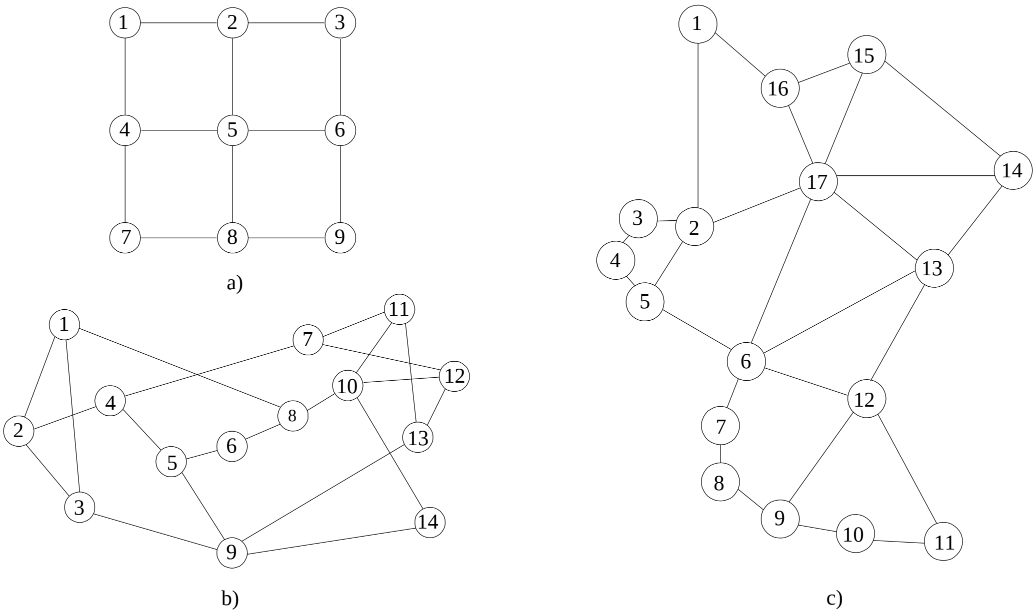


Figure 4.1: Network topologies: a) 9-node network, b) NSFNET 14, c) Germany 17.

simulation-based approach to get statistical results.

We first generate traffic patterns with lightpaths uniformly distributed among all node-pairs. The total number of lightpaths in each network ranges from 15 to 35 lightpaths. Obviously, with four wavelengths multiplexed in an optical fiber, the network cannot carry more lightpaths. We then use an LP-solver, in this case CPLEX 9.1, to solve the RWA problem for these traffic patterns in both cases: full wavelength conversion and no wavelength conversion. The mathematical models to solve these RWA problems can be found in [RS95] . All the traffic patterns, which are infeasible for the case of no wavelength conversion but feasible for the case of full wavelength conversion, will be used for our experiment. Clearly, only in this case, we need to place wavelength converters. For 9-node, 14-node and 17-node network, we found 148, 394 and 288 traffic patterns out of 500 randomly generated traffic patterns, respectively. The wavelength converter placement problem for these traffic patterns is solved by the optimization model introduced in Section 4.1.1 with CPLEX. The number of in/out ports is set to 4, meaning that each converter can support four conversions. The converters have full range conversion. Each node-pair has a set of three alternative shortest routes. These routes are pre-computed.

For each traffic pattern, the converters' positions are recorded. They are summed up at the end to get the number of appearances of converters at each node for the whole simulation. The results are reported in Fig. 4.2, 4.3 and 4.4 for three networks 9-node, 14-node and 17-node, respectively. As can be seen from the results, some nodes have a significant high number of appearances and dominate the others.

The 9-node network has a symmetric topology but the converter distribution is not symmetric. Node 2 and node 4 are basically similar in the network but they have significant difference of wavelength converter distribution. One may wonder the reason for this behaviour. This is because of the routing table calculation. Only three shortest routes are considered, and our routing calculation program tends to put the node with the lower index into a route. For example, three routes connecting node-pair (1-9) are (1-2-3-6-9), (1-2-5-8-9) and (1-2-5-6-9). Therefore, node 2 appears in more routes than node 4 and consequently, it has a higher probability of being equipped with wavelength converters. This again confirms that the routing has a significant impact on converter placement. With a given routing strategy, there is a correspondent converter placement scheme.

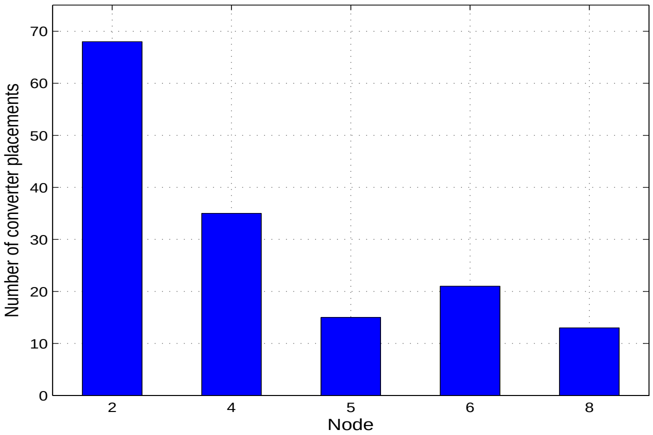


Figure 4.2: Wavelength converter distribution for 9-node network

4.1.2.2 Spreading or concentrating wavelength converters

Assuming that we have a certain number of wavelength converters, a question arises: should we place them at the node with highest probability of being equipped with converters or place them at several nodes with high probability?

To answer this question, we do an experiment for the 9-node network. We assume to have 3 converters, each of which can do only one conversion (in/out port =1). These

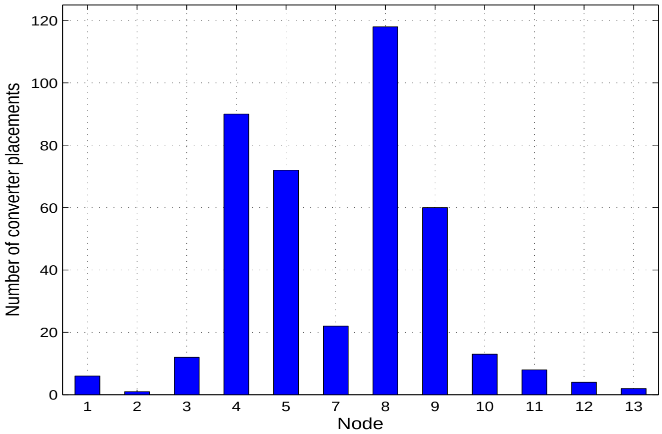


Figure 4.3: Wavelength converter distribution for NSFNET 14 network

converters can be distributed among three nodes 2, 4 and 5. 150 random traffic patterns have been generated. For each placement scheme, we apply our optimization model to calculate the percentage of traffic patterns that need additional converters. This refers to infeasibility (blocking) situation. The results are shown in Table (4.1).

Table 4.1: Efficiency of different converter placement schemes

Converter positions	Infeasibility probability
2	48.6%
$2-4$	28.4%
$2-5$	36.5%
$2-4-5$	19.6%

From the results, we see that, the infeasible cases is significantly reduced, from 48,6% to 19,6%, if we spread the converters to three nodes. We can conclude that, with the same number of wavelength converters, it is better to distribute them among some nodes, which have high probability of being equipped with converters, rather than to place all of them at just one node.

4.1.2.3 Complexity and application

The problem of RWA was proved to be NP-complete. So this problem is also NP-complete and hence it can be only applied for small networks or real-sized networks with low number

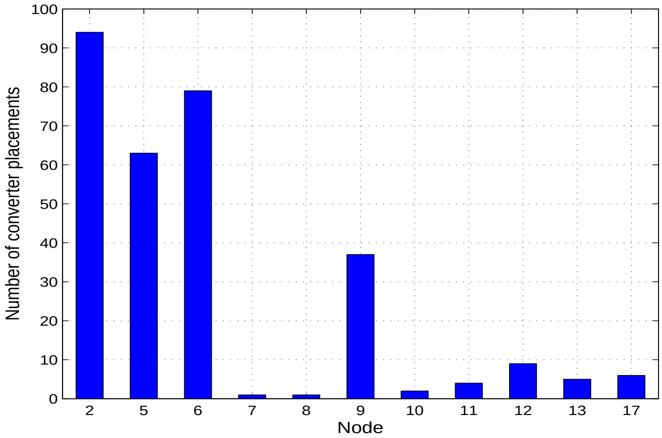


Figure 4.4: Wavelength converter distribution for Germany 17 network

of wavelengths. Real transmission systems normally have a large number of wavelengths multiplexed (8, 16, 40) and hence, applying the ILP formulation presented in Section 4.1.1 to the systems with a large number of wavelengths may not be pratical. However, to solve the wavelength converter placement problem, it is not necessary to solve the optimization problem for these real systems.

Clearly, we need the wavelength conversion only when the wavelength resource is sparse. Therefore, if there are more wavelengths multiplexed in an optical fiber, more lightpaths can be carried in the network, but the number of lightpaths that need wavelength conversions likely remains the same. Hence, the number of wavelengths does not affect the wavelength converter distribution. So for a real-network, we can assume there is only a small number of wavelengths in the network, and do the experiment as described in Section 4.1.2 to find nodes with high probability of being equipped with wavelength converters. This information is helpful for network planning, when the traffic is just roughly estimated and can change over time.

Moreover, our approach can work with different routing strategies. No fixed-routing is assumed as in many previous studies.

Another advantage of this approach is that it can solve the problem for different traffic demand distributions. In the experiment above, we assumed that the traffic demands are uniformly distributed among all node-pairs. However, in reality this is usually not true. Traffic is often concentrated in few nodes. In this case, we just need to generate traffic randomly corresponding to the real traffic distribution, do the statistical experiment and get the results accordingly. This overcomes the weakness of many algorithms, in which the assumption of uniform traffic distribution must hold.

4.2 Wavelength assignment in WDM networks with sparse limited-range wavelength converters

Wavelength assignment is a sole feature of WDM networks that other conventional networks do not have. Due to the wavelength contunuity requirement, wavelength assignment becomes non-trivial, and thus, a good algorithm is required in order to set up a lightpath succesfully whereas utilizing wavelength resources efficiently. For static traffic, wavelength assignment problem can be formulated as a graph-colouring problem [Muk97]. But for dynamic traffic, a heuristic algorithm is needed to minimize the blocking probability.

The wavelength assignment problem for dynamic traffic has been intensively studied but most of the studies do not take into account the sparse presence of wavelength converters in the network, which moves the wavelength assignment problem to another context. Since the number of wavelength converters is limited, a lightpath should use as few converters as possible to save converters for upcoming lightpaths. [ZY07] and [HL06] proposed similar algorithms called Longest-Segment (LS) and First-longest-lambda-run algorithm (FLR), respectively to deal with this problem. The basic idea of this algorithm is as follows: Starting from the source node, one looks for the longest sub-path that does not require a wavelength converter. From the end node of the sub-path, the process repeats until reaching the destination node. This algorithm is prooved to provide the optimal wavelength assignment in case that wavelength converters have full range conversion capability. In fact, wavelength converters may not be able to convert wavelength to any other wavelength. This refers to limited-range conversion. With this characteristic of wavelength converters, the well-known algorithm First-Fit and the beforementioned LS (FLR) become inefficient.

Let us consider an example depicted in Fig. 4.5. This is a given path with available wavelengths on it. Our task is to find a wavelength channel for this path. We assume that all converters on the path have the conversion range of 1, meaning that a wavelength can be converted to its closest wavelength on both sides. If First-Fit algorithm is used, from node 1 to node 9, wavelength 1 will be assigned. From node 9 onward, one cannot find an available wavelength because the converter at node 9 cannot convert wavelength w_1 to w_3 due to the limited-range conversion, resulting in a blocking decision. Similarly, LS algorithm also block the request because it cannot find available wavelengths for the given path. In fact, a wavelength channel exists as follows: from node 1 to 6: w_1, from node 6 to 9: w_2, from node 9 to 13: w_3 and from node 13 to 14: w_4. Therefore, a better wavelength assignment algorithm should be developed to overcome this problem.

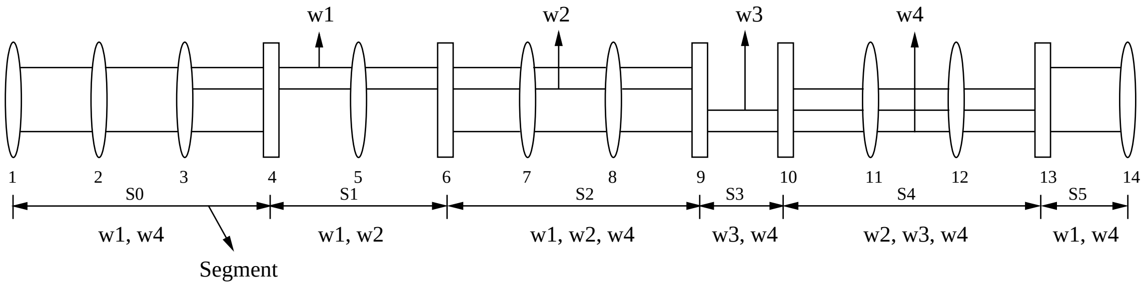


Figure 4.5: The state of a link-path with 14 nodes and 4 wavelengths. Wavelength converting nodes are shown as a rectangle

In this section, we introduce a distributed algorithm to assign wavelengths for a given path in case of sparse limited-range wavelength converters. By constructing a so-called Wavelength Tree, the algorithm minimizes the number of converters required for a lightpath.

4.2.1 Algorithm

We first introduce the algorithm from the centralized point of view. Its distributed implementation will be presented later.

4.2.1.1 Algorithm description

The algorithm starts with dividing the path into segments $S = S_0, S_1, ...S_i, ...S_K$, with K being the number of converters along the paths (excluding the converters placed at the source and destination nodes). A segment is defined as a sub-path, which starts with either the source node or a wavelength convertible node and ends with either the destination node or the next wavelength convertible node. An example of segments is shown in Fig 4.5. For each segment, we find the list of all available wavelengths on the whole segment $W_{S_k} = \{w_1, w_2, ...\}$.

We then construct the Wavelength Tree (WT) as follows. The source node is considered as the root node of the tree. The tree is composed of $K+1$ layers $\{L_0, L_1, ...L_i, ...L_K\}$, where L_i corresponds to segment S_i. Nodes at each layer are available wavelengths on that segment. These are called wavelength nodes. Each node has a label that represents the number of converters used from the source node to the end node of the corresponding segment. All nodes on the layer L_0 are connected to the root node and labeled with 0. Wavelength nodes on layer from L_1 to L_K will sequentially search for their parent nodes and calculate their labels according to the following rules.

- A wavelength node at layer L_i has the parent node at layer L_{i-1}. The wavelength represented at the parent node and the child node must be in the conversion range of each other.

- A node takes the same label as its parent node if it represents the same wavelength as its parent. This means there is no wavelength conversion needed between these two segments.

- A node has the label of its parent node plus 1 if it represents a different wavelength from its parent. This means there is a wavelength conversion between these two segments.

- A node has the label ∞ if it cannot find a parent node.

- A node chooses its parent node so that its label value is minimized. If there are multiple candidate parent nodes, which result in the same label value, the node with the lowest wavelength index will be chosen. This is because the First-Fit wavelength assignment algorithm, which was described in Section 2.2.1, always gives lower blocking probability than a random wavelength assignment algorithm [ZJM00].

Once the WT is constructed, we take the wavelength node with the lowest label value at the layer L_K and traverse back to the root node. This path shows how wavelengths are assigned on each segment.

4.2.1.2 Illustrative example

We consider a given path as shown in Fig. 4.5. Only free wavelengths are shown in the figure. There are five converters dividing the path into six segments. We assume that the conversion range is 1, meaning that a wavelength can be converted to its adjacent ones on both sides. The lists of all available wavelengths on each segment are as follows: $S_0\{w_1, w_4\}$, $S_1\{w_1, w_2\}$, $S_2\{w_1, w_2, w_4\}$, $S_3\{w_3, w_4\}$, $S_4\{w_2, w_3, w_4\}$, $S_5\{w_1, w_4\}$. The wavelength tree is then constructed as in Fig (4.6). This results in the wavelength assignment in each segment from the source to the destination as follows $W = \{w_1, w_1, w_2, w_3, w_3, w_4\}$.

In this example, as discussed above, both First-Fit and LS (FLR) algorithms failed to find available wavelengths for the requested lightpath.

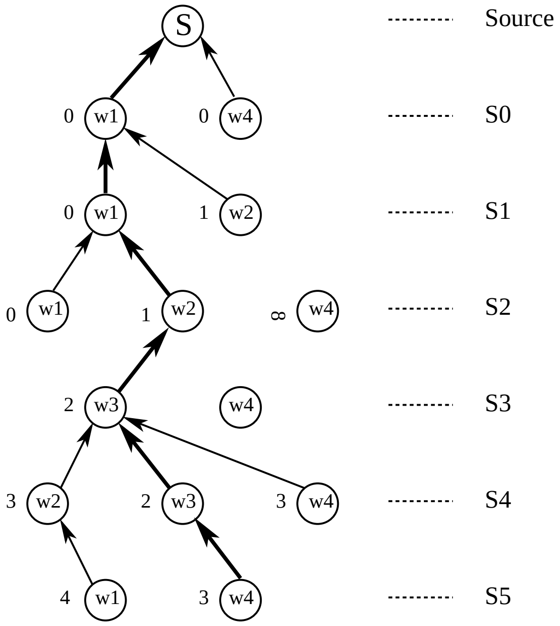


Figure 4.6: Visualization of the construction of the Wavelength Tree (WT)

4.2.1.3 Distributed implementation

A significant advantage of this proposed algorithm is that it can be implemented in a distributed manner with low communication overhead. As can be seen, the WT is constructed sequentially from layer L_0 to layer L_K. Therefore no global information about available wavelengths is needed and hence, we can implement the WT construction in a distributed manner as follows. Firstly, the source node sends out a message along the path to collect information regarding available wavelengths. Once this message reaches a convertible node, this node will construct the tree based on the received message and then sends the message containing the constructed tree to the next convertible node. With the information about available wavelengths and the currently constructed tree, a convertible node can continue to build up the tree and send it further. The destination node finishes the tree construction, analyses the WT and chooses the optimal path and sends the reservation message back to the source node. Once the source node receives the feedback from the destination, the wavelength channel is succesfully created and ready to transmit data. To set up a lightpath, the source node always has to send a request message to the destination node and the destination must send the confimation message back to the source. So basically, the distributed implementation of our proposed algorithm does not inccur any additional message exchanges.

4.2.1.4 Complexity

Let K be the number of wavelength converters, W be the number of wavelengths and R be the conversion range of wavelength converters. At each wavelength converting node, one has to scan through all wavelengths, and for each wavelength, wavelengths in its conversion range are examined to construct the Wavelength Tree. So the complexity of the algorithm is $\mathcal{O}(KWR)$. According to [ZY07], the complexity of LS (FLR) is $\mathcal{O}(KW)$. When the conversion range R decreases, the complexity of the proposed algorithm converges to the one of LS (FLR) algorithm.

4.2.2 Performance evaluation

We carried out an experiment with single-fiber NSFNET network shown in Fig 4.7. Each fiber contains 16 wavelengths. Five nodes (marked with black colour) are equipped with share-per-node wavelength converters. Each of these nodes has 16 availables converters, which results in a total 80 of converters in the network. Arrivals and holding time of requested lightpaths are assumed to follow a Poisson distribution and negative exponential distribution, respectively. Lightpath requests are uniformly distributed among node-pairs.

We first compare the blocking probability of our algorithm with the best existing one,

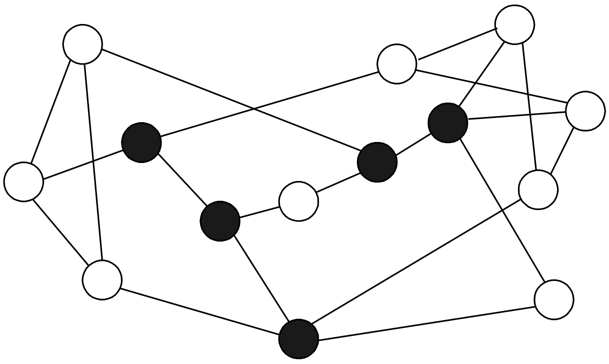


Figure 4.7: NSFNET network - 5 nodes equipped with wavelength converters

LS (or FLR) algorithm. We use the fixed-alternate routing scheme described in Section 2.1.1 with 5 alternative routes for each node pair and the routing strategy is shortest-path first. Fig 4.8 shows the blocking probability as a function of conversion range for different network loads. Our algorithm obviously results in a lower blocking probability compared to LS (FLR) algorithm when the conversion range is low. When the conversion range comes near to the full range conversion, both algorithms give more or less the same blocking probability. This is because the LS algorithm does not give the optimal solution in case of limited range conversion but also results in an optimal solution in case of full range conversion.

So to summarize, when the conversion range is small compared to the number of wavelength, it is recommended to use our algorithm because it offers a better blocking probablity and comparable complexity. Especially in case the conversion range is 1, our algorithm has the same complexity as LS (FLR) algorithm but results in an obviously better performance. When the conversion range is large or even full, it is recommended to use LS (FLR) algorithm.

We also examine the proposed algorithm for two different routing strategies:

- Shortest path first (SPF): among all sorted alternative routes, the first available shortest route is chosen. This strategy saves wavelength resources.
- Least-converter first (LCF): among all sorted alternative routes, the route resulting in lowest number of wavelength converters is chosen. This strategy saves converters.

We run a simulation for these two routing strategies in two cases: one with 32 con-

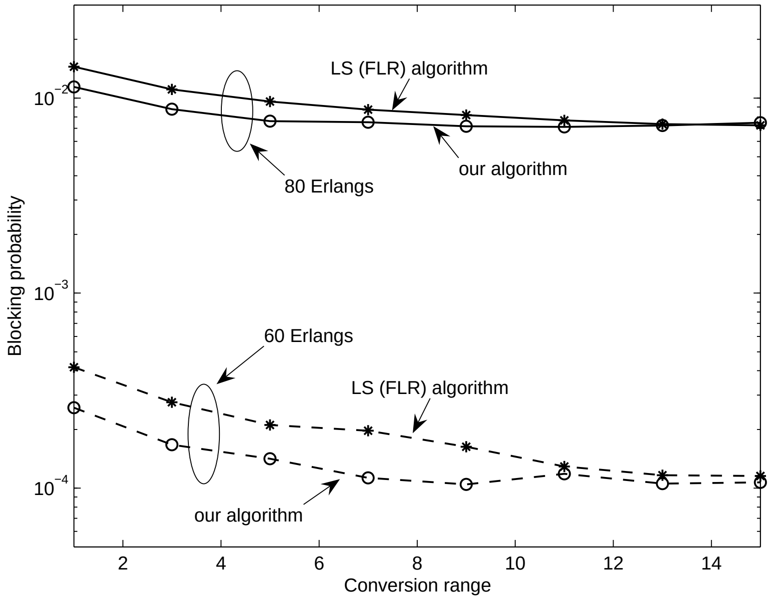


Figure 4.8: Blocking probability vs. conversion range

verters at each convertible node, meaning abundant converter resources, one with only 8 converters at each node, which means scarcity of converters. The conversion range is 4 in this simulation. Results are shown in Fig 4.9.

For both cases, LCF has slightly better performance in terms of blocking probability. One may expect in the case of plentiful converters, the SPF routing strategy should have better performance because it uses less wavelength resources. But this is not the case. The reason is that if we use less converters for a lightpath, we somehow maintain the wavelength continuity in the network. This is important for networks with limited-range converters because converters may become useless if wavelengths on two links are not in conversion range of each other. This again confirms the importance of minimizing the number of wavelength conversion in WDM networks with limited-range wavelength converters.

4.3 Conclusion

In this chapter, we studied the problem of wavelength converter placement and wavelength assignment in WDM networks with sparse limited-range wavelength converters.

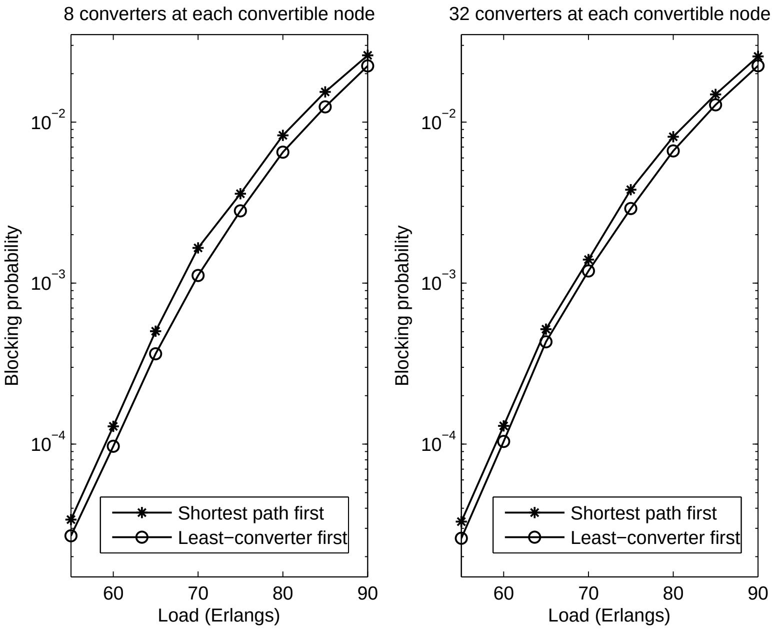


Figure 4.9: Blocking probability vs. load (Erlangs)

First, an exact ILP formulation was introduced to solve the wavelength converter placement problem for static traffic patterns, which are infeasible if no wavelength converter is installed but feasible in case of full wavelength converters. The advantage of this approach is that the RWA problem and wavelength converter placement problem are considered at the same time. With the exact ILP formulation, we can solve the RWA and wavelength converter placement for small networks. Due to the complexity of the problem, it cannot be applied for large networks with a huge number of wavelengths. However, with the statistical method, we can in any case find the nodes offering the highest pay-back when equipped with wavelength converters.

In addition, a distributed algorithm for wavelength assignment was introduced. The algorithm is based on a WT, which is constructed sequentially by wavelength converting nodes along the path, to obtain the optimal wavelength assignment with the least number of conversions. A significant advantage of this algorithm is that it can be implemented in a distributed manner without additional message exchanges whereas offering a lower blocking probabilty to the best existing one. Another finding is that in networks with limited range converters, it is more important to save converters to maintain the wavelength continuity than to save wavelength resources.

Chapter 5

Logical Topology Design for IP over WDM Networks

This chapter discusses the logical topology design problem, using IP over WDM, a two-layer network architecture, as a reference network. In wavelength-routed networks, many lightpaths are created to carry traffic and to support different kinds of services at the higher layer. The nodes together with the set of lightpaths form a logical topology, on which IP traffic is transported. It may not be possible to set up lightpaths between any two nodes due to insufficient number of wavelengths and limited number of transceivers. Therefore, designing a logical topology is important and challenging. It requires a good solution to efficiently utilize network resources whereas guaranteeing network performance such as network congestion or propagation delay. The logical topology design problem consists of several subproblems such as determining the topology, routing and wavelength assignment of lightpaths on the physical layer and routing IP traffic on the logical layer. Solving these problems jointly is challenging because of its high complexity, but solving them separately may lead to a sub-optimal solution. Therefore, a method of logical topology design that provides a good solution and at the same time is not so complex to be used for large networks is needed to effectively exploit all valued features of WDM networks.

In this chapter, a new approach to design logical topologies is introduced. First, we formulate the problem as an MILP problem with our proposed objective function. A numerical example for small networks is presented to investigate the performance of the proposed approach. Since the MILP is NP-hard and thus cannot not be applied to large networks, we then employ the iterative optimization algorithm proposed in Chapter 3 to solve logical topology design problem for real-sized networks. This work was reported in [TK07a, TK07b, TK08b]

5.1 Problem statement

While designing logical topologies with the objective of optimizing network performance in terms of lightpaths' utilization, propagation delay or network throughput, like in [MBRM96, RS96, KS01, BM00, DR00, ZVZO03, LST06, LS05], we observe that the resulting network tends to use all resources to set up as many lightpaths as possible. For example, in case of minimizing lightpaths' load, many lightpaths are set up so that traffic can be spread over these lightpaths to reduce lightpaths' load and thus avoid network congestion. In case of minimizing average hop distance, the network tends to set up a full-mesh topology if possible to reduce multi-hop communication to single-hop communication in order to reduce propagation delay. Similar behavior happens while maximizing network throughput. The consequence is that once the logical topology is set up, there will be no or very few remaining resources for future lightpaths. These approaches are suitable only when only one logical topology exists on the WDM network and the logical topology planning is long-term.

In reality, there may be more than one logical topology implemented on top of a WDM network. These topologies can belong to different Internet Service Providers (ISPs), different Virtual Private Networks (VPN) or different services. They have to share the same network resources, i.e. wavelengths and transceivers. Therefore, while designing a logical topology for a specific traffic pattern, one should save resources for future use. Moreover, in real networks, traffic demands of a node-pair change considerably over time. Reconfiguration of logical topologies is therefore required to guarantee network performance. If all resources are used for a current topology, moving to a new topology will definitely cause disruption because no new lightpaths can be set up before deleting at least one existing lightpath. This problem becomes much more critical in optical networks due to the large bandwidth of a wavelength channel. A large amount of data will be disrupted when a lightpath is torn down and service disruption is very expensive. Consequently saving resources for a reconfiguration process is also necessary.

We therefore want to design a logical topology so that the resources are used most efficiently. We observe that for a two-layer network architecture like IP-over-WDM, minimizing the number of set-up lightpaths or used wavelength-links [10] like in [JBA06] is not enough to ensure resource efficiency. A lightpath between two nodes can be set up if and only if there is a pair of transceivers (transmitter and receiver) at these nodes and an available wavelength channel connecting them. Minimizing the number of lightpaths results in a low number of used transceivers but maybe a large number of wavelength-links. Similarly, minimizing the number of wavelength-links may lead to inefficient transceiver usage. Consequently, it can happen that some nodes have free transceivers but no free wavelengths on links connected to them. Likewise, there may be available wavelengths but no transceivers in order to set up a lightpath, and thus, the useful free resource is reduced. This refers to blocking situations. To overcome this problem, we propose a new approach to design logical topologies. Our approach is to formulate the problem as an MILP problem with the objective of maximizing the number of remaining lightpaths whereas taking the network performance, such as lightpaths load or hop distance as a constraint. Our proposal results in the best resource utilization.

5.2 MILP approach

The problem of designing logical topologies is formulated as a mixed-integer linear programming (MILP), using the following notations:

- s and d denote source and destination nodes of an IP packet, respectively.
- i and j denote originating and terminating nodes of a lightpath, respectively.
- m and n denote endpoints of a physical link.
- $W = \{k\}$: set of wavelengths multiplexed in an optical fiber, where k denotes wavelengths.
- $R = \{r\}$: set of alternative routes, where r denotes route candidates for a lightpath in physical layer.

---

[10] A wavelength-link indicates a wavelength on a link

5.2.1 Given parameters

- Physical topology: $P = \{f_{mn}\}$, where f_{mn} indicates the number of fibers connecting nodes m and n, and $f_{mn} = f_{nm}$.
- $\delta^{ij}_{mn,r}$ denotes the existence of link mn in route r connecting node-pair (i, j). If $\delta^{ij}_{mn,r} = 1$, link mn belongs to route r between nodes (i, j) and $\delta^{ij}_{mn,r} = 0$ otherwise.
- Traffic matrix $H = \{h_{sd}\}$, where h_{sd} denotes the traffic demand between the source destination pair (s, d).
- T_i and R_i denote the number of transmitters and receivers, respectively, at node i.
- Network congestion indicator: L_{max} denotes the allowed maximum load in a lightpath. L_{max} should be lower than the capacity of a wavelength.

5.2.2 Variables

- Logical topology $Y = \{y_{ij}\}$, where y_{ij} is the number of lightpaths between node i and j. A lightpath is not necessarily bidirectional. Hence, $y_{ij} \neq y_{ji}$.
- Traffic routing: t^{sd}_{ij} denotes the traffic between source and destination nodes (s, d), routed through logical link ij.
- Physical routing: $p^{ij}_{r,\lambda}$ denotes the number of times the wavelength k is assigned to route r connecting node-pair (i, j). In single-fiber networks (where $f_{mn} \leq 1$), $p^{ij}_{r,k}$ becomes a binary variable.
- x_{ij} denotes the the number of remaining lightpaths connecting node-pair (i, j).

5.2.3 Constraints

- Degree constraints:

$$\sum_j y_{ij} \leq T_i \qquad , \forall i, j \tag{5.1}$$

$$\sum_i y_{ij} \leq R_j \qquad ,\forall i, j \tag{5.2}$$

The above constraints guarantee that the number of lightpaths originating from and terminating at a node is constrained by the number of transmitters and receivers at that node, respectively.

– Traffic constraints at logical topology:

$$\sum_j t_{ij}^{sd} - \sum_j t_{ji}^{sd} = \begin{cases} h_{sd} & s = i \\ -h_{sd} & d = i \\ 0 & s \neq i, d \neq i \end{cases} \qquad ,\forall s, d \tag{5.3}$$

$$\sum_{sd} t_{ij}^{sd} \leq L_{max} \cdot y_{ij} \qquad ,\forall i, j \tag{5.4}$$

Equation 5.3 is a multi-commodity flow equation representing the traffic routing on logical links. Here, we assume that the traffic is bifurcated. Equation 5.4 ensures that the load on each lightpath does not exceed a pre-determined maximum load L_{max}, which represents network congestion. Basically L_{max} can be equal to the wavelengths capacity. However, it is not recommended to have a logical topology with some lightpaths having the utilization of 100%. We therefore suggest that L_{max} be in the range from 50% to 70% of a wavelength capacity, depending on network requirements. Other constraints related to network performance can be added, such as delay bound or hop-count bound, if necessary. In the scope of this work, we omit it.

– Constraints at physical layer:

$$\sum_r \sum_\lambda p_{r,\lambda}^{ij} = y_{ij} \qquad ,\forall i, j \tag{5.5}$$

$$\sum_{ij} \sum_r p_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} \leq f_{mn} \qquad ,\forall m, n, \lambda \tag{5.6}$$

Constraint 5.5 ensures that all lightpaths of the logical topology can be routed in the physical topology. In case of sparse resources, this constraint may not be satisfied, resulting in an infeasible solution for the logical topology design problem. Constraint 5.6 ensures that the number of times a wavelength is used on a link is lower than or equal to the number of fibers on that link, so that a wavelength is used only once in each fiber, avoiding wavelength conflict.

– Calculating possible remaining lightpaths:

To calculate the number of remaining lightpaths, we introduce the variable $z_{r,\lambda}^{ij}$ representing the number of times the wavelength λ is assigned to route r connecting node-pair (i, j). This variable is similar to the variable $p_{r,\lambda}^{ij}$, which represents the routing and wavelength assignment of a lightpath between node i and j. In single-fiber network, $z_{r,\lambda}^{ij}$ becomes a binary variable. This variable is similar to the variable $p_{r,\lambda}^{ij}$

$$\sum_{j} x_{ij} \leq T_i - \sum_{j} y_{ij} \qquad , \forall i \tag{5.7}$$

$$\sum_{i} x_{ij} \leq R_j - \sum_{i} y_{ij} \qquad , \forall j \tag{5.8}$$

$$x_{ij} = \sum_{r} \sum_{\lambda} z_{r,\lambda}^{ij} \qquad , \forall i, j \tag{5.9}$$

$$\sum_{ij} \sum_{r} z_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} \leq f_{mn} - \sum_{ij} \sum_{r} p_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} \qquad , \forall m, n, \lambda \tag{5.10}$$

Constraint 5.7 and 5.8 restrict the number of remaining lightpaths to the number of remaining transmitters and receivers, respectively. Constraint 5.9, similar to constraint 5.5, ensures that all remaining lightpaths can be realized at the physical layer. Constraint 5.10, ensures that the number of times a wavelength used on a link for remaining lightpaths does not exceed the remaining wavelength resources.

5.2.4 Objective

The objective of this formulation is to maximize the number of remaining lightpaths, which can be set up simultaneously.

$$\max \sum_{ij} x_{ij} \tag{5.11}$$

Using this objective function, we observed that the optimization process tends to count as many short free lightpaths as possible to increase the objective value. Its advantage is to guarantee the good arrangement of free transceivers and wavelengths so that the number of useless free transceivers and wavelengths is minimized. However, in reality, we also need to create longer lightpaths. To do that, it is necessary to have some free wavelengths on the whole network. This also helps to increase the possibilities to create different lightpaths. Future created lightpaths are normally unknown beforehand, so increasing the possibilities to set up a lightpath can reduce the blocking probability. The following illustrative example will give a clearer explanation.

For a given logical topology in Fig. 5.1(a), there are two different solutions for wavelength assignment in the physical topology as shown in Fig. 5.1(b)and (c). The numbers on each link show the wavelengths used on that link. We assume that there are 4 wavelengths in each fiber and 4 transceivers at each node. In both solutions, we can set up additionally 8 different lightpaths simultaneously (two lightpaths on each link). However, the solution in Fig. 5.1(c) is preferred because in the absence of wavelength converters, it offers possibilities to set up a lightpath between any node-pair while the one in Fig. 5.1(b) does not.

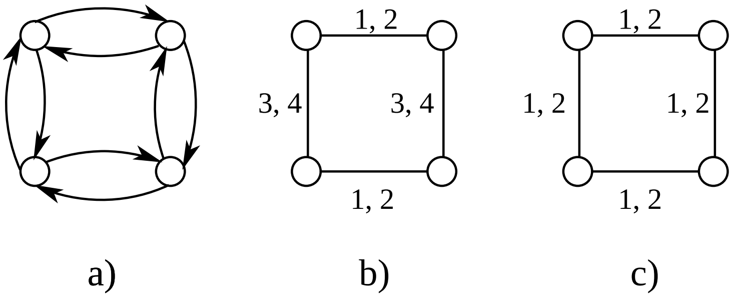


Figure 5.1: Routing and wavelength assignment example

To solve the problem above, we change the objective function slightly. We first assign an increasing cost to each wavelength $Cost(W) = \alpha_\lambda$, where W is the number of wavelengths multiplexed in each optical fiber, λ is the wavelength index, and $\alpha_{\lambda+1} > \alpha_\lambda$. The problem is then solved with a new objective function.

$$\max \sum_{ij} x_{ij} - \theta \cdot \sum_{\lambda} \alpha_\lambda \cdot \sum_{ij} \sum_{r} p^{ij}_{r,\lambda} \cdot \delta^{ij}_{mn,r} \tag{5.12}$$

where θ is chosen small enough so that the added term does not affect the main objective function in 5.11. This objective function ensures that wavelengths with lowest index will be used first and hence saves some wavelengths in the whole network.

5.2.5 Additional constraints

The constraints presented in Section 5.2.3 are sufficient for the formulation of our problem. However, in this section, we present some additional constraints that narrow the solution space without excluding the optimal solution. They therefore may help to speed up the optimization process. These constraints are as follows:

$$t_{ij}^{sd} \leq y_{ij} \cdot h_{sd} \qquad , \forall sd, ij \tag{5.13}$$

$$\sum_{d} h_{sd} \leq L_{max} \cdot \sum_{j} y_{sj} \qquad , \forall s \tag{5.14}$$

$$\sum_{s} h_{sd} \leq L_{max} \cdot \sum_{i} y_{id} \qquad , \forall d \tag{5.15}$$

Constraint 5.13 ensures that the traffic only flows through existing lightpaths. Constraints 5.14 and 5.15 give the lower bound for the number of lightpaths going out and coming in an IP node.

5.3 Heuristic approach

The MILP problem introduced above is NP-hard. It therefore becomes intractable for real-size networks. For 14-node networks, CPLEX failed to find a feasible solution after 20 hours of running-time, under Linux with processor of 3.2 GHz, 512 MB RAM. To cope with this problem, we use a heuristic algorithm, the so-called iterative optimization proposed in Chapter 3.

5.3.1 Outline of heuristic algorithm

Obviously the node-pair with high traffic demand should be given the priority to have a direct lightpath connecting nodes. Lower traffic volumes can be groomed with other traffic and may pass through several lightpaths before reaching the destination. According to this observation, the heuristic algorithm will accommodate traffic based on its rank. The heuristic algorithm is presented in the flow-chart in Fig. 5.2.

The algorithm starts by sorting all traffic demands in deceasing order. At each step, we choose N highest traffic demands that are not yet accommodated, relevant to a small number of M nodes to form a sub-traffic matrix. All other entries of this sub-traffic matrix are set to 0. M is determined beforehand, based on the capability of the MILP solver. The reason to choose traffics relevant to a small number of nodes is to reduce the problem from large networks to small networks. Therefore the problem can be solved within a reasonable time. It is acceptable to do that because in real networks, traffic flows are normally concentrated around some nodes, rather than evenly distributed among all nodes.

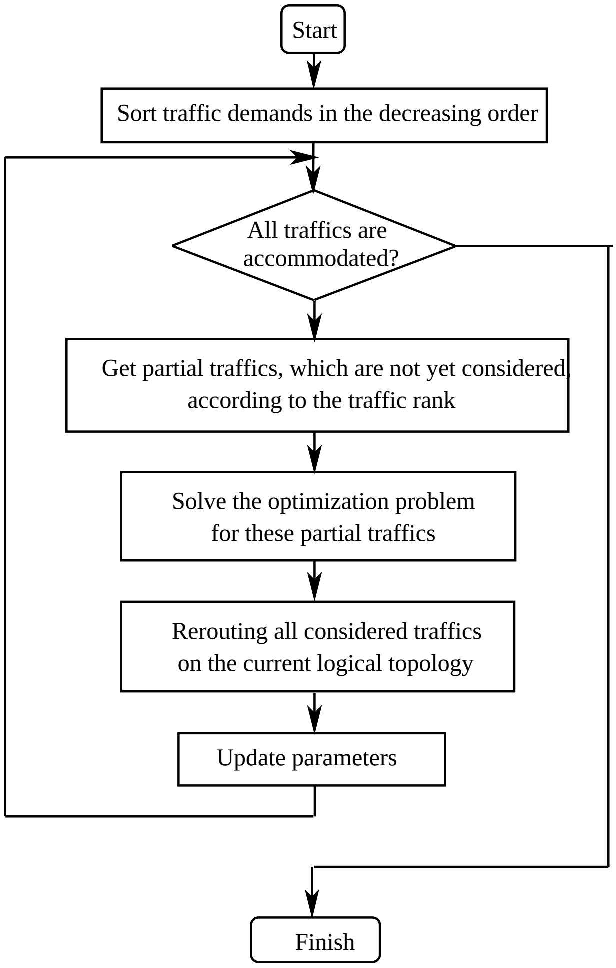


Figure 5.2: Heuristic algorithm diagram

It may happen that the N highest traffics relate to more than M nodes. In this case, we just choose the highest traffic relevant to M nodes, skip the other traffics and continue with the highest traffics among the selected M nodes. In the next iteration, we go back to the highest traffics that are not yet accommodated, and choose N highest ones relevant to other M nodes. It also may happen that we can not always find exactly N traffics relevant to M nodes in every iteration, because some traffics among those M nodes are already accommodated in previous iterations. In this case, we just consider all available traffics among M nodes. Once the partial traffic matrix is formed, the optimization model is applied and solved for this matrix. After each optimization process, some new lightpaths can be created to form an intermediate logical topology. Since there are some new lightpaths, it might be better if some traffic demands considered in previous iterations pass through these new lightpaths. Therefore, after the optimization process, all considered traffic is re-routed on the new logical topology. The purpose of this step is to minimize the total traffic on the logical network. After re-routing, network parameters, such as the number of transceivers, wavelengths and so on are updated for the next iteration. The process is repeated until all traffic demands are accommodated. Clearly, the number of traffic demands considered at each iteration will affect the results, both in terms of quality as well as time-consumption.

5.3.2 Optimization formulation for each iteration

After each iteration, the numbers of set-up lightpaths, lightpaths load and resources (transceivers, wavelengths) are changed. Hence, compared to the formulation introduced in the Section 2, the optimization formulation for each step must be slightly changed. The exact optimization formulation is shown as follows. Besides parameters and variables defined in Section 5.2.1 and Section 5.2.2, we introduce some new parameters:

- Existing set-up lightpaths e_{ij}: the number of lightpaths between node i and j that are already created.
- L_{ij}: total traffic passing through lightpaths ij.
- $q_{mn,\lambda}$: the number of times that wavelength λ is used on link mn. For single-fiber network, $q_{mn,\lambda}$ becomes a binary parameter.

The new formulation will be as follows:

$$\sum_j y_{ij} \leq T_i - \sum_j e_{ij} \qquad ,\forall i \tag{5.16}$$

$$\sum_i y_{ij} \leq R_j - \sum_i e_{ij} \qquad ,\forall j \tag{5.17}$$

Constraints 5.16 and 5.17 ensure that new lightpaths can only be set up between nodes i and j if there are still free transceivers between them.

The multi-commodity flow Eq. 5.3 is kept the same.

$$\sum_{sd} t_{ij}^{sd} \leq L_{max} \cdot (y_{ij} + e_{ij}) - L_{ij} \qquad ,\forall i,j \tag{5.18}$$

Constraint 5.18 ensures that traffic passing through a lightpath does not exceed the predetermined lightpaths load L_{max}. $L_{max} \cdot (y_{ij} + e_{ij}) - L_{ij}$ represents the available capacity in lightpaths between i and j. Besides new lightpaths y_{ij}, traffic can pass through existing lightpaths e_{ij} if they still have available bandwidth.

$$\sum_{ij} \sum_r p_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} \leq f_{mn} - q_{mn,\lambda} \qquad ,\forall mn, \lambda \tag{5.19}$$

Constraint 5.19 ensures that there are free wavelengths for new lightpaths.

To calculate the number of remaining lightpaths, constraints 5.7, 5.8, 5.9 and 5.10 are modified as follows.

$$\sum_j x_{ij} \leq T_i - \sum_j y_{ij} - \sum_j e_{ij} \qquad ,\forall i \tag{5.20}$$

$$\sum_i x_{ij} \leq R_j - \sum_i y_{ij} - \sum_i e_{ij} \qquad ,\forall j \tag{5.21}$$

Constraint 5.9 is kept the same here.

$$\sum_{ij} \sum_r z_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} \leq f_{mn} - \sum_{ij} \sum_r p_{r,\lambda}^{ij} \delta_{mn,r}^{ij} - q_{mn,\lambda} \qquad ,\forall mn, \lambda \tag{5.22}$$

The additional constraints 5.13, 5.14, and 5.15 are replaced by:

$$t_{ij}^{sd} \leq (y_{ij} + e_{ij}) \cdot h_{sd} \qquad ,\forall sd, ij \tag{5.23}$$

$$\sum_{d} h_{sd} \leq L_{max} \cdot \sum_{j} (y_{sj} + e_{sj}) \qquad ,\forall s \tag{5.24}$$

$$\sum_{s} h_{sd} \leq L_{max} \cdot \sum_{i} (y_{id} + e_{id}) \qquad ,\forall d \tag{5.25}$$

Obviously, the complexity of this formulation is the same as the one in Section 5.2. However, since for each step, only a small number of non-zero traffic demands relevant to a small number of nodes are considered, the number of non-zero integer variables is therefore much reduced. Hence, the problem becomes a small problem and we can obtain the result within a reasonable time.

5.4 Performance evaluation

In this section, the performance of the proposed logical topology design approach is evaluated. We first apply the MILP model to a small network to evaluate the performance of the proposed approach against the existing ones. Later, the performance of the heuristic algorithm is evaluated using a 14-node (NSFNET) network. Finally, we compare the MILP solution with the solution obtained by the heuristic algorithm.

5.4.1 Numerical examples for a small network

To see the efficiency of our formulation compared to other logical topology design approaches, we use the single-fiber 6-node network in Fig. 5.3 as an illustrative numerical example. We assume that there are 4 transceivers at each node and 4 wavelengths multiplexed in a fiber.

5.4.1.1 Comparision with other approaches

This section visualizes the difference between our approach with the one minimizing the number of set-up lightpaths and the one minimizing the number of wavelength-links. For

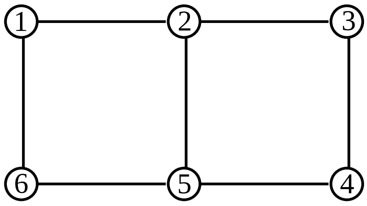


Figure 5.3: 6-node network topology

this purpose, a traffic matrix, of which entries are chosen randomly from a uniform distribution in (0, 1), was generated as shown in Table 5.1. The capacity of a wavelength is assumed to be 2.5Gbps. L_{max} is chosen to be 1.8Gbps, corresponding to 72% lightpath utilization.

Table 5.1: Traffic matrix for 6-node network

0.000	0.537	0.524	0.710	0.803	0.974
0.391	0.000	0.203	0.234	0.141	0.831
0.060	0.453	0.000	0.645	0.204	0.106
0.508	0.660	0.494	0.000	0.426	0.682
0.480	0.174	0.522	0.879	0.000	0.241
0.950	0.406	0.175	0.656	0.193	0.000

To minimize the number of set-up lightpaths, constraints 5.7, 5.8, 5.9, and 5.10 are waived and the objective function is replaced by:

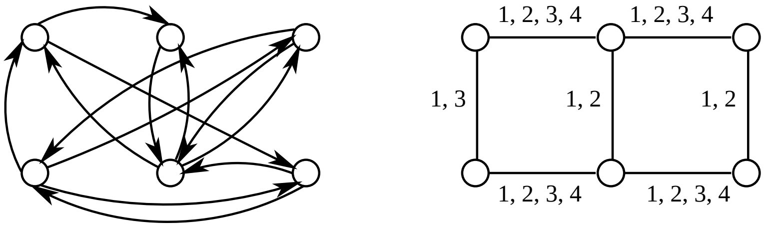
$$\min \sum_{ij} y_{ij} + \theta \cdot \sum_{\lambda} \alpha_{\lambda} \cdot \sum_{ij} \sum_{r} p_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} \tag{5.26}$$

Similar to our objective function in 5.12, the term $\theta \cdot \sum_{\lambda} \alpha_{\lambda} \cdot \sum_{ij} \sum_{r} p_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij}$ is added to save wavelengths and also to ensure fairness while comparing with our approach.

To minimize wavelength capacity, the objective function is replaced by:

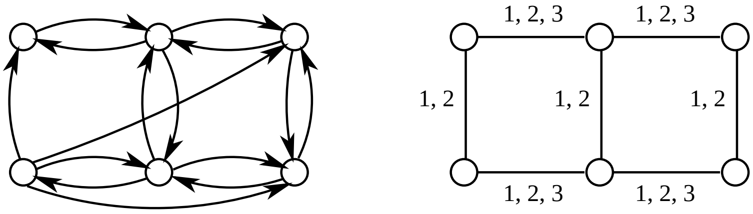
$$\min \sum_{\lambda} \alpha_{\lambda} \cdot \sum_{ij} \sum_{r} p_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} + \theta \cdot \sum_{ij} y_{ij} \tag{5.27}$$

The term $\theta \cdot \sum_{ij} y_{ij}$ is added to ensure that among all possible optimal solutions, we choose the one with lowest number of set-up lightpaths, which is of course the most



(a) Logical topology (b) RWA in physical layer

Figure 5.4: Minimizing the number of setup lightpaths



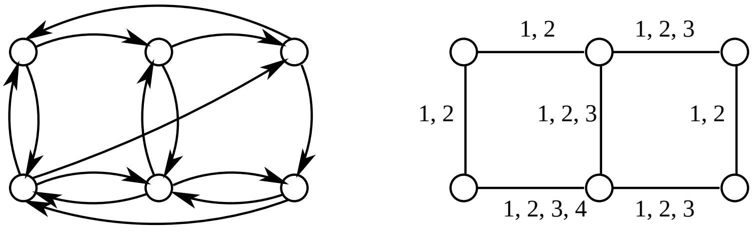
(a) Logical topology (b) RWA in physical layer

Figure 5.5: Minimizing the number of wavelength-links

preferred solution. In any case, θ is chosen small enough so that the additional term does not affect the main objective of minimizing the number of wavelength-links.

The results of these three approaches are shown in the Fig. 5.4, Fig. 5.5, and Fig. 5.6. The number on each link shows the wavelength used on that link.

Minimizing the number of lightpaths results in 13 lightpaths and 22 wavelength-links. Minimizing the number of wavelength-links results in 18 wavelength-links but 15 lightpaths in total. In the former case, some long lightpaths are created and hence more wavelength-links are needed. Consequently, there are only 6 free lightpaths at maximum. Even worse, there are 4 links with full wavelength utilization and therefore, some node-pairs have no possibility to create a lightpath. The latter case is better with 9 remaining lightpaths. However, the node at the bottom-left corner (Fig. 5.5a) has four outgoing lightpaths, meaning all transmitters at this node are used and there is no possibility to create a lightpath from this node. Our approach (Fig. 5.6) results in 14 lightpaths, 19 wavelength-links and 10 remaining lightpaths. Moreover, there are some free transceivers at every node. Hence, we have possibility to create a lightpath between any node-pair. Obviously, our approach brings the best solution in terms of resource utilization while the network



(a) Logical topology (b) RWA in physical layer

Figure 5.6: Maximizing the number of remaining lightpaths

performance is the same in all three approaches.

5.4.1.2 Some more results

Table 5.2: Results for traffic matrix 1 - 6-node network

L_{max}	$\min \sum W$	$\min \sum Lightpaths$	$\max \sum Remaining \quad lightpaths$
1.5	8 - 16 - 18	9 - 15 - 19	9 - 15 - 19
1.6	9 - 15 - 18	6 - 14 - 22	9 - 15 - 18
1.7	10 - 14 - 18	9 - 14 - 19	10 - 14 - 18
1.8	9 - 15 - 18	6 - 13 - 24	10 - 14 - 19

Table 5.3: Results for traffic matrix 2 - 6-node network

L_{max}	$\min \sum W$	$\min \sum Lightpaths$	$\max \sum Remaining \quad lightpaths$
1.5	5 - 17 - 22	4 - 16 - 24	6 - 18 - 22
1.6	5 - 16 - 22	4 - 15 - 24	6 - 18 - 22
1.7	7 - 16 - 20	6 - 15 - 22	8 - 16 - 20
1.8	7 - 15 - 19	6 - 14 - 22	9 - 15 - 19

We carried out some more experiments to see the performance of the proposed formulation. We run the optimization for three different randomly generated matrices with different values of L_{max}. The entries of traffic matrices are chosen randomly from [0, 1] with uniform distribution. These traffics are assumed to have unit of Gbps. The results are shown in the Table 5.2, Table 5.3, and Table 5.4. In each cell in the table, one sees three numbers. The first represents the number of remaining lightpaths, which can be created simultaneously. The second is the number of set-up lightpaths for the designed logical

Table 5.4: Results for traffic matrix 3 - 6-node network

L_{max}	$\min \sum W$	$\min \sum Lightpaths$	$\max \sum Remaining \quad lightpaths$
1.5	6 - 18 - 21	5 - 16 - 23	6 - 18 - 21
1.6	6 - 17 - 20	7 - 16 - 21	7 - 17 - 20
1.7	7 - 16 - 20	7 - 15 - 21	8 - 16 - 20
1.8	7 - 16 - 19	7 - 15 - 19	8 - 16 - 19

topology. The last is the total number of wavelength-links, which are assigned to the set-up lightpaths.

From the tables above, we see that minimizing the number of lightpaths results in lowest number of set-up lightpaths but in most of the cases, to obtain it, long lightpaths are created and hence more wavelength resources are needed. Minimizing the number of wavelength-links results in lowest used wavelength resource but may need to create more lightpaths. The common problem of those two approaches is that they do not guarantee the best arrangement of free transceivers and wavelengths. Some free resources therefore become useless. In many cases, one sees that the number of created lightpaths and the number of wavelength-links is the same as in our approach but only maximizing remaining lightpaths will offer the largest number of free lightpaths.

5.4.2 Numerical example for a real-sized network

A numerical example for the heuristic algorithm is generated based on NSFNET single-fiber network, which contains 14 nodes and 21 links. We assume that there are 16 wavelengths multiplexed in a fiber and 16 transceivers at each node. In the WDM layer, alternative routing strategy is applied. Five shortest routes are pre-selected between every node-pair.

We run the experiments for three different traffic matrices, with different values of L_{max} and different number of traffics considered in one iteration. The number of relevant nodes per iteration is 6. The first traffic matrix is taken from [MBRM96] shown in Table 5.5. It corresponds to a measured traffic distribution of NSFNET backbone network. Two other traffic matrices are randomly generated. The second is created by picking 5 node-pairs at random and assigning an amount of traffic uniformly distributed in the range $[0; 200]$ for both directions. Ten other node-pairs are allocated traffic from $[0; 20]$ and others are

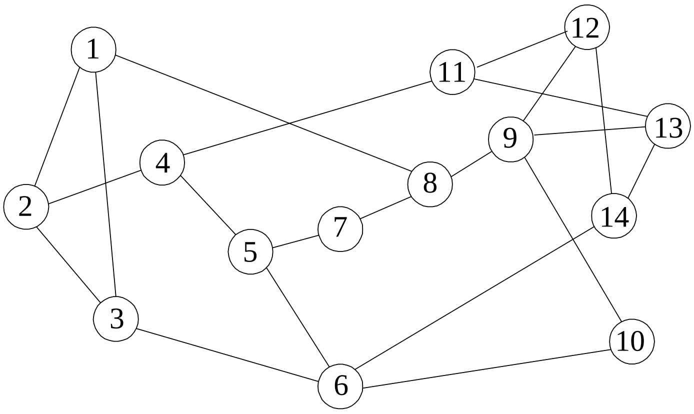


Figure 5.7: 14-node NSFNET topology

allocated a random amount of traffic chosen from a uniform distribution in $[0; 1]$. The third matrix is created similarly with 6 node-pairs having high traffics uniformly distributed in $[0; 200]$, 12 node-pairs having medium traffics from $[0; 20]$. These values are relative, so their unit is not important.

Table 5.5: Measured traffic matrix of NSFNET backbone network, taken from [MBRM96]

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0.00	1.09	2.06	0.14	0.45	0.04	0.43	1.45	0.51	0.10	0.07	0.09	0.00	0.33
2	11.71	0.00	8.56	0.62	11.12	7.77	3.62	15.79	3.66	16.61	2.03	37.81	4.83	13.19
3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	0.31	3.41	13.64	0.00	1.90	0.60	0.70	2.88	2.00	3.26	3.07	6.69	0.08	4.01
5	0.28	67.51	19.02	3.43	0.00	4.03	10.77	62.22	24.02	17.92	0.45	79.03	9.97	5.29
6	0.00	5.81	3.42	5.52	3.4	0.00	2.61	2.68	0.87	3.87	0.04	0.84	0.06	2.48
7	1.75	22.02	102.31	4.47	22.03	7.9	0.00	114.10	19.82	21.95	0.78	71.40	0.33	32.84
8	2.39	63.84	210.30	8.52	28.21	2.66	97.08	0.00	43.95	33.00	11.37	48.63	5.53	13.85
9	6.45	18.93	37.35	6.0	24.99	6.81	25.06	61.02	0.00	39.62	14.52	127.50	23.34	0.76
10	0.05	35.29	10.26	3.73	22.34	9.48	4.98	57.08	6.84	0.00	6.30	17.64	5.91	0.76
11	0.10	1.02	3.13	1.69	0.24	0.06	0.81	1.45	0.58	7.12	0.00	0.84	0.06	0.50
12	1.28	26.15	1.00	5.94	24.86	1.32	5.49	40.57	29.53	22.37	10.50	0.00	1.01	0.54
13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
14	0.73	29.09	13.63	9.89	35.61	12.07	6.44	28.79	4.67	0.00	3.99	0.00	10.75	0.00

The results for these three traffic matrice are shown in Table 5.6, Table 5.7, and Table 5.8, respectively. Similar to the results shown in Table 5.2, each cell has three numbers representing the number of remaining lightpaths, created lightpaths and wavelength-links, respectively. LP-relaxation [11] solutions are also reported for comparison purpose. In this

---

[11] LP-relaxation is the problem that arises by removing integer constraints of all variables. This relax-

case, LP-relaxtion yealds the upper bound for the number of remaining lightpaths.

As can be seen from the results, the quality of the heuristic algorithm varies with the number of considered traffic demands in one iteration. However its trend is not clear but depends on the traffic patterns. For traffic matrix 1 and 3, 15 traffics per iteration gives the best performance, while for traffic matrix 2, 6 traffics per iteration tends to be the best. This is due to the restriction of relevant nodes in each iteration. When the traffic is highly concentrated in several nodes, choosing a large number of traffics per iteration tends to give better performance because it ensures that high traffic demands will be accommodated first. But when the traffic is more widely spread, choosing high number of traffics per iteration is not good because some small traffic demands might be accommodated before the high ones, which then results in worse performance.

Table 5.6: Results for traffic matrix - NSFNET network

L_{max}(Gbps)	Traffic demands/iteration			LP Relaxation
	6	10	15	
125	66 45 85	71 41 73	69 43 81	83.0988
150	72 40 72	74 37 72	76 36 65	86.6067
175	75 36 58	76 36 65	77 34 62	89.2335

Table 5.7: Results for traffic matrix 2 - NSFNET network

L_{max} (Gbps)	Traffic demands/iteration			LP Relaxation
	6	10	15	
125	81 30 56	81 30 58	84 28 57	93.1773
150	84 27 55	81 31 60	83 29 57	95.0611
175	86 26 48	80 32 60	84 28 56	96.3734

While the performance vs. the number of traffics per iteration is not the same in all cases, the time to get the solution has a clear trend. Table 5.9 shows the average time to get the solution vs. the number of traffics per iteration.

Even though a large number of traffics per iteration causes the lower number of iterations, it is still clear that the higher the number of traffics considered in one iteration,

---

ation technique transforms an NP-hard optimization problem into a related problem that is solvable in polynomial time. The solution to the relaxed problem can be used to gain information about the solution to the original program.

Table 5.8: Results for traffic matrix 3 - NSFNET network

L_{max}(Gbps)	Traffic demands/iteration			LP Relaxation
	6	10	15	
125	78 33 65	79 - 33 - 56	79 33 64	91.7553
150	81 31 53	83 29 60	82 30 64	93.8376
175	82 29 51	84 28 49	85 27 52	95.2614

Table 5.9: Time to get solution

Traffics/iterations	Average time (s)
6	471
10	886
15	3349

the longer the time to get the solution. This is because the time to solve the optimization problem for a large number of traffics is much higher than in the case of a low number of traffics per iteration. Therefore, even though there is less number of iterations, a large number of traffics per iteration still takes longer time to get the final solution. According to the results in Table 5.6, Table 5.7, and table 5.8, it is not necessary that a higher number of traffics per iteration offers the better performance. So we can choose an appropriate number of traffics for an iteration so that we can get a good solution within a reasonable time.

5.4.3 Comparison of heuristic and MILP approaches

In this section, we evaluate the performance of the heuristic approach against the MILP approach. We use the 6-node network in Fig. 5.3, with the same parameters as in the numerical example in Section 5.4.1, to compare two approaches, because MILP can be applied only to small networks. Numerous traffic matrices are randomly generated with the value uniformly distributed in the range of $[0; 1]$. For each traffic matrix, we solve the problem using these two approaches with different values of L_{max}. In the heuristic approach, two cases, 6 traffic demands per iteration and 15 traffic demands per iteration, are considered. The average number of remaining lightpaths in each approach corresponding to each L_{max} is calculated and shown in Fig. 5.8. The solving time is also recorded. The

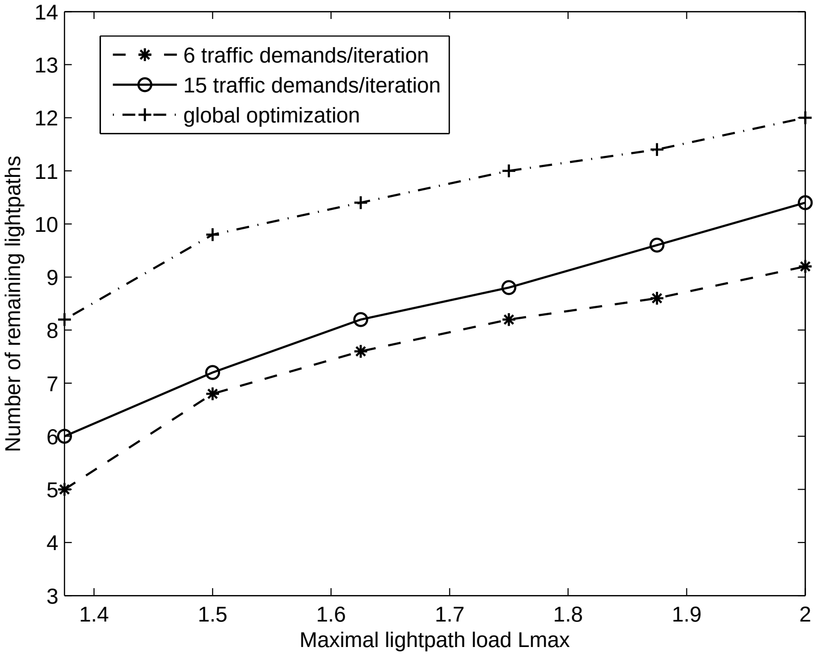


Figure 5.8: Average number of remaining lightpaths vs. L_{max}

average solving time is shown in Fig. 5.9.

Obviously, the results of heuristic approaches are worse than that of the MILP approach but the time to solve the problem reduces dramatically. Therefore, for large networks, it is necessary to use the heuristic algorithm.

5.5 Conclusion

In this chapter, the logical topology design problem for IP over WDM networks was studied. A new design approach was proposed to enhance the efficiency of the network resource utilization. The problem was first formulated as an MILP problem. Different from all other approaches, our objective function is to maximize the number of remaining lightpaths while network performance (in this thesis, i.e network congestion represented by maximal lightpaths' load) is guaranteed by constraints. This formulation avoids the blocking situation of the other two approaches: minimizing the number of lightpaths and minimizing the number of wavelength-links. The numerical examples confirmed that the proposed formulation

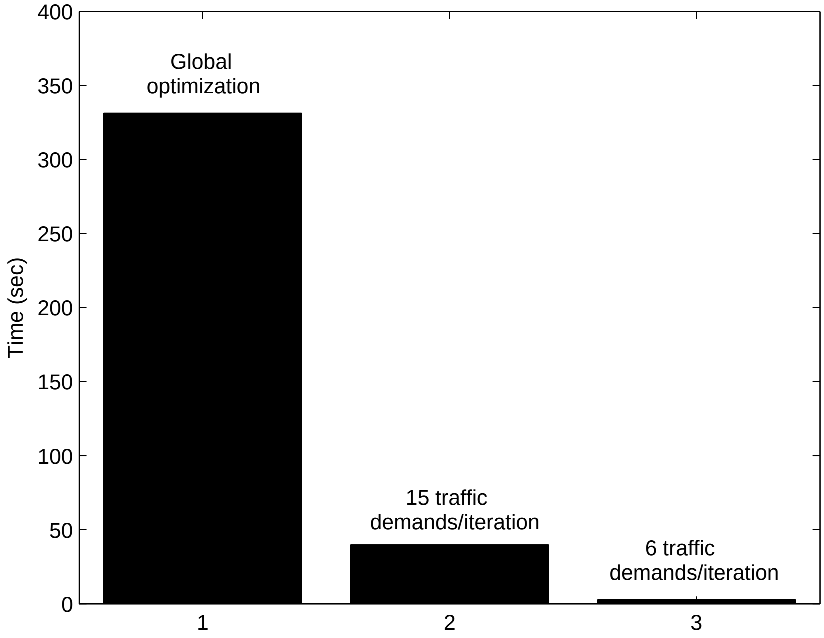


Figure 5.9: Time to solve the problem

results in the best resource efficiency.

The MILP problem is NP-hard and rapidly becomes intractable for large networks. Hence, a heuristic algorithm based on the iterative optimization algorithm described in Chapter 3 was proposed to solve the problem for large networks. The heuristic approach was developed by dividing the problem into sub-problems. These sub-problems were solved iteratively and their results are integrated to get the solution to the original problem. The heuristic guarantees to get a good solution in a reasonable time. The quality of the heuristic algorithm depends on the size of the sub-problems. However, there is no strong correlation between these two variables. For different traffic patterns, the quality of the heuristic algorithm may increase or decrease with the size of sub-problems. In any case, the time to get the solution decreases when the size of sub-problems decreases.

Chapter 6

Logical Topology Reconfiguration under Dynamic Traffic

As studied in the previous chapter, a logical topology can be optimally designed regarding some objective function for a specific traffic pattern. However, as traffic demands are subject to change over time, logical topologies may be outdated and hence cannot carry the IP traffic efficiently. In this case, the logical topology needs to be changed in response to changing traffic. The process of changing a logical topology to meet traffic requirements is called reconfiguration. The ability to reconfigure logical topologies is a key feature of wavelength-routed networks. Reconfiguration of logical topology is a non-trivial task. It does not only require a good selection of a new topology to efficiently carry IP traffic, but also a smooth migration from the old topology to the new one to avoid service disruption, which is very expensive.

In this chapter, we study the logical topology reconfiguration problem for two-layer architecture networks, IP over WDM, under dynamic traffic. The problem is solved with two approaches: centralized and distributed. In the centralized approach, it is formulated as ILP problem and solved by optimization tools. This mathematical model is highly complex and hence can be applied to small networks only. Therefore, we introduce an approximate model, which simplifies the selection process of changing lightpaths, so that the model can be applied to larger networks. Afterwards, the Lagrangian relaxation method [12] is employed to decompose the approximate model into sub-problems which can be solved separately

---

[12] Lagrangian relaxation is a relaxation technique which works by moving hard constraints into the objective in order to execute a penalty on the objective if they are not satisfied.

at each node. This forms a distributed algorithm, which is the significance of this work compared to existing ones. The distributed algorithm provides a sub-optimal solution but different from the centralized approaches of which the complexity increases exponentially with problems' size, its complexity only increases linearly with the problem's size, and hence, it can be scaled to large networks. In addition, the distributed algorithm can lend itself to a protocol implementation. This work was reported in [TK08c, TK09].

6.1 Problem definition

6.1.1 Network model

We consider a backbone network of N nodes connected by one or multiple optical fibers forming a multi-fiber mesh optical network. Each fiber has W multiplexed wavelengths. Each node i is equipped with T_i transmitters and R_i receivers. We assume there is no wavelength converter in the network. Lightpaths are unidirectional. A lightpath can be established between two nodes if there is a transmitter/receiver pair at these nodes and a free wavelength channel connecting them.

Each node is capable of periodically measuring the load on all lightpaths originating from it. The information about free wavelengths on each fiber is also available at the node connected to it. All nodes in the network can exchange messages using link-state routing protocol. We assume that the control plane works perfectly, meaning that there is no loss of signalling messages during the reconfiguration process.

Backbone traffic is the aggregation of several end-systems and hence the short term variations are filtered out. The traffic on the network is therefore observed to change smoothly over time and to repeat in day periods. In this study, we used the traffic computed in the GEANT network [UQBL06] as the traffic model. The traffic pattern is available every 15 minutes for a period of 4 months. We therefore assume that the measurement of lightpaths' load is done every 15 minutes. An implicit assumption here is that the time for reconfiguration is much shorter than the measurement period. In addition, we presume that all nodes have synchronized clocks so that they measure the lightpaths' load at the same time.

6.1.2 Design goals

In a reconfiguration process, the new logical topology must be optimized in terms of some specific objectives. Moreover, it is necessary to guarantee no service disruption during reconfiguration. Service disruption is very critical in optical networks due to the huge capacity of wavelength channels. It is also necessary to use resources efficiently. This is important because spare resources are necessary for the transition process from an old to a new logical topology. In case there is no spare resource, no new lightpath can be added before deleting one or several lightpaths, which then will cause network disruption. At last, it is desirable that the lifetime of new topologies is as long as possible so that the number of reconfigurations during a given period could be reduced. The goals of our algorithm can be summarized as follows:

- Network performance must be guaranteed.
- No service disruption occurs during reconfiguration process.
- Resources are utilized efficiently.
- The number of reconfigurations for a long period is minimized.

6.1.3 Overview of the algorithm

We introduce three load parameters: upper bound T_{high}, lower bound T_{low} and *load balance indicator* L ($T_{low} < L < T_{high}$). A lightpath is considered to be overloaded if its load exceeds T_{high} and underutilized if its load is lower than T_{low}. Each node periodically measures the load of all lightpaths originating from it. Whenever an overloaded or under-utilized lightpath is detected, the node will broadcast the reconfiguration-triggering message to all other nodes. The reconfiguration process is then started.

The reconfiguration will result in a new logical topology with lowest number of lightpaths in such a way that the maximum lightpath load at the time of reconfiguration is restricted to the value of the *load balance indicator* L. The idea of limiting lightpaths' load at the value of L, with $L < T_{high}$, is to give some room for traffic variation so that the lightpaths' load will not violate T_{high} right after the reconfiguration. Consequently, the life-time of the new logical topology is extended. To ensure no service disruption, we always add all new lightpaths first, and re-route the traffic through the new logical topology before deleting any lightpaths. [BEP$^+$96] also added all new lightpaths first to avoid network disruption but in their study, they assumed the new logical topology was known beforehand and therefore they required sufficient resources for the union of old and new topologies. Different from [BEP$^+$96], in our study, we search for the new topology based on the old topology and free resources.

Our algorithm allows adding and deleting several lightpaths at the same time. Different from all existing algorithms, our algorithm also decides if it is really necessary to add/delete lightpaths or just do rerouting on the current logical topology. The lifetime of logical topologies is investigated based on the load balance indicator L. In our study, two subproblems, selecting added/deleted lightpaths and migrating to the new topology, are carried out at the same time as a joint optimization.

6.2 Centralized approach

In this approach, we introduce two mathematical models to solve the reconfiguration problem. The first model solves the problem in an exact manner. The second model is an approximate model, which simplifies the exact model by removing some constraints to reduce the complexity. The second model will be the base to develop a distributed algorithm. These models are formulated as an ILP problem, using the following notations:

- s and d denote the source and destination nodes of IP packets, respectively.
- i and j denote the originating and terminating nodes of a lightpath, respectively.
- m and n denote endpoints of a physical link.
- $W = \{\lambda\}$, set of wavelengths.
- $R = \{r\}$: set of alternative routes, where r denotes route candidates for a lightpath at physical layer.

6.2.1 Exact mathematical model

6.2.1.1 Given parameters

- Physical topology: $P = \{f_{mn}\}$, where f_{mn} indicates the number of fibers connecting node m and n and $f_{mn} = f_{nm}$.

- $q_{mn,\lambda}$ denotes the number of times the wavelength has been used on link mn by the current logical topology. In the case of single fiber networks, $q_{mn,\lambda}$ becomes a binary variable.

- $\delta^{ij}_{mn,r}$ denotes the existence of link mn in route r connecting node-pair (i, j). If $\delta^{ij}_{mn,r} = 1$, link mn belongs to route r connecting node-pair (i, j) and $\delta^{ij}_{mn,r} = 0$ otherwise.

- Traffic matrix $H = \{h_{sd}\}$, where h_{sd} denotes the traffic rate between source-destination pair (s, d).

- Current logical topology: $Y = \{y_{ij}\}$, where y_{ij} is the number of lightpaths between node i and j. A lightpath is not necessarily bidirectional. Hence $y_{ij} \neq y_{ji}$.

- Load balance indicator L: the maximum lightpath load of the new logical topology at reconfiguration point must be smaller than or equal to L.

6.2.1.2 Decision variables

- Added lightpaths: a_{ij} denotes the number of lightpaths added between node i and node j.

- Deleted lightpaths: b_{ij} denotes the number of lightpaths deleted between node i and node j.

- Traffic routing: t^{sd}_{ij} denotes the traffic between source destination pair (s, d), routed through new logical link ij.

- Physical routing for added lightpaths: $p^{ij}_{r,\lambda}$ denotes the number of times the wavelength λ is assigned to route r connecting node-pair (i, j). In single-fiber networks, $p^{ij}_{r,\lambda}$ is a binary variable.

6.2.1.3 Constraints

– Transceiver constraints:

$$\sum_j (y_{ij} + a_{ij}) \leq T_i \qquad , \forall i \tag{6.1}$$

$$\sum_i (y_{ij} + a_{ij}) \leq R_j \qquad , \forall j \tag{6.2}$$

The term $(y_{ij} + a_{ij})$ represents the union of current and new logical topology. Because we always add all new lightpaths before deleting any lightpaths, the above constraints guarantee that the number of existing and being added lightpaths originating from and terminating at a node is constrained by the number of transmitters and receivers at that node respectively.

– Traffic constraints on logical topology:

$$\sum_j t_{ij}^{sd} - \sum_j t_{ji}^{sd} = \begin{cases} 1 & s = i \\ -1 & d = i \\ 0 & s \neq i, d \neq i \end{cases} \qquad , \forall s, d \tag{6.3}$$

$$\sum_{sd} t_{ij}^{sd} \cdot h_{sd} \leq L \cdot (y_{ij} + a_{ij} - b_{ij}) \qquad , \forall i, j \tag{6.4}$$

$$b_{ij} \leq y_{ij} \qquad , \forall i, j \tag{6.5}$$

Equation (6.3) is a multi-commodity flow equation representing the traffic routing on logical links. We assume that traffic is non-bifurcated, and thus, t_{ij}^{sd} is a binary variable. The number of lightpaths connecting nodes i and j is represented by $(y_{ij} + a_{ij} - b_{ij})$. Hence, equation (6.4) guarantees that the maximum lightpath's load in the new logical topology is smaller than or equal to L. Equation (6.5) ensures that only existing lightpaths can be deleted.

– Constraints in physical layer:

$$\sum_{r}\sum_{\lambda} p_{r,\lambda}^{ij} = a_{ij} \qquad ,\forall i, j \tag{6.6}$$

$$\sum_{ij}\sum_{r} p_{r,\lambda}^{ij} \cdot \delta_{mn,r}^{ij} \leq f_{mn} - q_{mn,\lambda} \qquad ,\forall mn, \lambda \tag{6.7}$$

Equation (6.6) represents the routing and wavelength assignment for all added lightpaths. The right-hand side of the constraint (6.6) represents the free wavelength resources. All added lightpaths are set up using those free resources. Constraint (6.7) ensures that the number of times a wavelength is used on a link is lower than or equal to the number of fibers on that link so that a wavelength is used only once in each fiber.

6.2.1.4 Objective

$$\min \sum_{ij} (a_{ij} - b_{ij}) + \alpha \cdot \sum_{ij} a_{ij} \tag{6.8}$$

The objective is to minimize the number of lightpaths used for the new logical topology, which is represented by the first term, as well as the total number of lightpath changes, which is represented by the second term. It is motivated by the following ideas. The first is to save resources and energy to operate the IP network. The second is to reduce the management cost while managing the logical topology. The third is that provided we restrict the maximum lightpaths' load to a value of L, minimizing the number of lightpaths will increase the minimum lightpaths' load and hence contribute to load balancing in the network. The last is to reduce the cost of reconfiguration. Because we give higher priority to the minimization of the number of lightpaths, a small co-efficient α is added to the second term so that it does not affect the first term. Besides minimizing the number of lightpaths in the new logical topology, 6.8 also tries to minimize the number of added lightpaths and thus, the total number of lightpath changes is minimized.

6.2.2 Approximate mathematical model

Solving the optimization problem described in the previous section for a real-size network (tens of nodes) is impossible since it is NP-complete. Therefore it is necessary to reduce the complexity to deal with larger networks. An approximate model is hence introduced to deal with medium-sized networks. The approximate model is later the base to develop the distributed algorithm.

The reconfiguration process starts with selecting some candidate lightpaths that are likely to be added or deleted. Instead of searching for all possibilities to add and delete lightpaths as in the optimization problem in Section 6.2.1, the approximate model just chooses added and deleted lightpaths among the candidate ones. The procedure of selecting candidate lightpaths is described at the end of this section.

6.2.2.1 Mathematical approximate model

Besides the parameters introduced in Section 6.2.1, we introduce two necessary parameters for this approximate model, i.e $cand\_a_{ij}$ and $cand\_b_{ij}$. Let $cand\_a_{ij}$ and $cand\_b_{ij}$ be the number of added and deleted candidate lightpaths of the node pair (i, j), respectively. The model introduced in Section 5.2 can be reduced as follows.

- Objective function:

$$\min \sum_{ij} (a_{ij} - b_{ij}) + \alpha \cdot \sum_{ij} a_{ij} \tag{6.9}$$

$$or \quad \min \sum_{ij} ((1+\alpha)a_{ij} - b_{ij}) \tag{6.10}$$

- Constraints:

$$\sum_j t_{ij}^{sd} - \sum_j t_{ji}^{sd} = \begin{cases} 1 & s = i \\ -1 & d = i \\ 0 & s \neq i, d \neq i \end{cases} \qquad ,\forall s, d \tag{6.11}$$

$$\sum_{sd} t_{ij}^{sd} \cdot h_{sd} \leq L \cdot (y_{ij} + a_{ij} - b_{ij}) \qquad ,\forall i, j \tag{6.12}$$

$$a_{ij} \leq cand\_a_{ij} \qquad ,\forall i, j \tag{6.13}$$

$$b_{ij} \leq cand\_b_{ij} \qquad ,\forall i, j \tag{6.14}$$

From the formulation above, we see that the added and deleted lightpaths are just chosen among some pre-determined candidate lightpaths. The constraints 6.6 and 6.7, which represent the routing and wavelength assignment of added lightpaths in the physical layer, are omitted. This is done explicitly while selecting candidate added lightpaths, using fixed-alternate routing strategy and First-Fit wavelength assignment algorithm mentioned in Section 2.2.1. The complexity of the optimization problem is therefore reduced and consequently it can be applied to larger networks.

6.2.2.2 Selection of added/deleted candidate lightpaths

Since not all lightpaths are considered in the reconfiguration process, the selection of candidate lightpaths is important. Choosing good candidate lightpaths will result in a good new logical topology and hence increases the performance of the algorithm.

The algorithm of selecting candidate lightpaths starts with sorting the new traffic demands in decreasing order. The selection process will be based on this traffic rank. The specific algorithm to select added and deleted lightpaths is presented as follows.

Let h_{sd}^{new} and h_{sd}^{old} be the traffic demands between node pair (s, d) at the time of current and previous reconfiguration, respectively. $cand\_a_{sd}$ and $cand\_b_{sd}$ are the number of added and deleted candidate lightpaths between node pair (s,d).

Selection of added candidate lightpaths

Assuming that the number of added candidate lightpaths is restricted to A lightpaths, the pseudo-code of the algorithm is shown in Algorithm 1.

Algorithm 1 Pseudo-code for selecting candidate added lightpaths

```
go through all traffic demands in the decreasing order
int $num\_add = 0$
int $cand\_a_{sd} = 0$
while $num\_add \leq A$ and not all demands are considered do
  if $h_{sd}^{new} - h_{sd}^{old} > 0$ then
    int $a = \lceil \frac{h^{new} - y_{sd} \cdot L}{L} \rceil$
    while $a > 0$ do
      if find a lightpath from $s$ to $d$ then
        $cand\_a_{sd} \leftarrow cand\_a_{sd} + 1$
        $num\_add \leftarrow num\_add + 1$
        $a \leftarrow a - 1$
      else
        $a \leftarrow 0$
      end if
    end while
  end if
end while
```

Selection of deleted candidate lightpaths

Assuming that the number of deleted lightpaths is restricted to B lightpaths, the pseudo-code for selecting candidate lightpaths is shown in Algorithm 2:

Algorithm 2 Pseudo-code for selecting candidate deleted lightpaths

```
go through all traffic demands in the increasing order
int $num\_del = 0$
$cand\_b_{sd} = 0$
while $num\_del \leq B$ and not all demands are considered do
  if $(h_{sd}^{new} - h_{sd}^{old} < 0)$ and $(y_{sd} > 0)$ then
    $cand\_b_{sd} \leftarrow cand\_b_{sd} + 1$
    $num\_del \leftarrow num\_del + 1$
  end if
end while
```

6.3 Distributed approach

Even though the approximate model is less complex than the exact one, it is still an NP-complete problem. Thus its solving time increases exponentially with the problem's size.

It is therefore not scalable to large networks. Hence, finding a sub-optimal solution within a reasonable time is the only choice so that the algorithm can be implemented online. The Lagrangian relaxation (LR) method has been successfully employed to solve complex optimization problems by means of constraint relaxation and problem decomposition. In this thesis, LR method is applied to solve the approximate ILP problem presented in Section 6.2.2. By using Lagrange multipliers, some constraints of the original problem are relaxed to form a dual problem. This dual problem is then decomposed into sub-problems, which are in turn solved independently by each node. The optimal solution of the dual problem is the lower bound for the original one. Because some constraints are relaxed in the dual problem, the solution of the dual problem is often an infeasible solution to the original problem. Therefore, we propose some heuristic adjustment to achieve a near-optimal feasible solution.

6.3.1 Lagrangian relaxation

We apply Lagrange relaxation to eliminate the capacity constraint (6.12) in the approximate ILP problem presented in Section 6.2.2. This constraint is moved to the objective function to form the Lagrangian associated with the original problem as follows:

$$\begin{aligned} L(a,b,t,\mu) &= \sum_{ij}((1+\alpha)a_{ij} - b_{ij}) + \sum_{ij}\mu_{ij}\left(\sum_{sd} t_{ij}^{sd} - L(y_{ij} + a_{ij} - b_{ij})\right) \\ &= f(a,b) + \sum_{ij}\mu_{ij}\left(\sum_{sd} t_{ij}^{sd} - L(y_{ij} + a_{ij} - b_{ij})\right) \end{aligned} \tag{6.15}$$

where Lagrangian multiplier $\mu = \{\mu_{ij}\}$ is a vector of non-negative scalars and $f(a,b)$ is the objective function in (6.10). Each μ_{ij} (or congestion pricing variable) can be considered as a penalty per unit violation of the capacity constraints. The Lagrangian dual function is:

$$q(\mu) = \min_{a,b,t \in F} L(a,b,t,\mu) \tag{6.16}$$

subject to Constraints 6.11, 6.13, and 6.14

The key property of the Lagrangian relaxation is that the minimization of the Lagrangian over the set of remaining constraints (denote this set F: constraints 6.11, 6.13, and 6.14) yields a lower bound for the optimal cost of the original problem. This is because:

$$\begin{aligned}\min_{a,b,t\in F} L(a,b,t,\mu) &\leq \min_{a,b,t\in F;\sum\limits_{sd} t_{ij}^{sd}-L(y_{ij}+a_{ij}-b_{ij})\leq 0} L(a,b,t,\mu)\\ &\leq \min_{a,b,t\in F;\sum\limits_{sd} t_{ij}^{sd}-L(y_{ij}+a_{ij}-b_{ij})\leq 0} \{f(a,b)\}\end{aligned}$$

The first inequality is because the minimization is taken over a subset of F, while the second inequality is due to the non-negativity of μ_{ij}.

Clearly, we are motivated to determine the tightest lower bound. This forms the dual problem of the primary problem.

$$\max\left\{q(\mu) = \min_{a,b,t\in F} L(a,b,t,\mu)\right\} \tag{6.17}$$

$$\text{subject to } \mu_{ij} \geq 0$$

6.3.1.1 Solving Lagrangian dual function

To solve the dual problem, we first consider problem (6.16). The Lagrangian can be rewritten as follows:

$$\begin{aligned}L(a,b,t,\mu) &= \sum_{ij}(1+\alpha)a_{ij} - \sum_{ij} b_{ij} + \sum_{ij}\mu_{ij}\sum_{sd} t_{ij}^{sd} - L\sum_{ij}\mu_{ij}y_{ij} - L\sum_{ij}\mu_{ij}a_{ij} + L\sum_{ij}\mu_{ij}b_{ij}\\ &= \underbrace{\sum_{ij}(1+\alpha-L\mu_{ij})a_{ij} + \sum_{ij}(L\mu_{ij}-1)b_{ij} - L\sum_{ij}\mu_{ij}y_{ij}}_{L_1(a,b,\mu)} + \underbrace{\sum_{ij}\mu_{ij}\sum_{sd} t_{ij}^{sd}}_{L_2(t,\mu)}\end{aligned} \tag{6.18}$$

The problem of minimizing the Lagrangian can be divided into two independent subproblems. The first problem is:

$$\min L_1(a,b,\mu) \tag{6.19}$$

subject to: $a_{ij} \leq cand\_a_{ij}$ and $b_{ij} \leq cand\_b_{ij}$

This is a simple problem. The solution is as follows:

$$\begin{aligned} a_{ij} &= \begin{cases} 0 & \text{if} \quad 1+\alpha \geq L\mu_{ij} \\ cand\_a_{ij} & \text{if} \quad 1+\alpha < L\mu_{ij} \end{cases} \qquad \forall i,j \\ b_{ij} &= \begin{cases} 0 & \text{if} \quad 1 \leq L\mu_{ij} \\ cand\_b_{ij} & \text{if} \quad 1 > L\mu_{ij} \end{cases} \qquad \forall i,j \end{aligned} \tag{6.20}$$

While knowing μ_{ij}, each node can calculate the added/deleted lightpaths originating from itself separatedly.

The second problem is:

$$\min L_2(t,\mu) = \sum_{ij} \mu_{ij} \sum_{sd} t_{ij}^{sd} = \sum_{sd} \sum_{ij} \mu_{ij} t_{ij}^{sd} \tag{6.21}$$

subject to:

$$\sum_j t_{ij}^{sd} - \sum_j t_{ji}^{sd} = \begin{cases} 1 & s = i \\ -1 & d = i \\ 0 & s \neq i, d \neq i \end{cases} \qquad \forall s,d \tag{6.22}$$

We can easily see that this is the shortest path problem for a non-capacitated network with μ_{ij} being the link weight. Each node can apply Dijkstra algorithm to find the shortest path from itself to the other nodes independently.

6.3.1.2 Solving dual problem

Since the dual problem (6.17) is non-continuous, it is solved by employing the sub-gradient method.We define the *subgradient* of $q(\mu)$ at a given $\mu^k > 0$ is a vector $g = \{g_{ij}\}$ such that:

$$q(\mu) \leq q(\mu^k) + (\mu - \mu^k)' \cdot g \qquad , \forall \mu \tag{6.23}$$

where $(\mu - \mu^k)'$ is the transpose of the matrix $(\mu - \mu^k)$.

From the formula of Lagrangian function (6.15), we can see that $g_{ij} = \sum_{sd} t_{ij}^{s} d - L(y_{ij}+$

$a_{ij} - b_{ij})$ is the subgradient of $q(\mu)$. This is because:

$$\begin{aligned}
q(\mu) &= \min_{a,b,t} \left(L(a,b,t,\mu) \right) \\
&\leq L(a,b,t,\mu) \\
&\leq f(a,b) + \sum_{ij} \mu_{ij} \left(\sum_{sd} t_{ij}^{s} d - L(y_{ij} + a_{ij} - b_{ij}) \right) \\
&\leq f(a,b) + \mu' \cdot g \\
&\leq f(a,b) + (\mu^k)' \cdot g + (\mu - \mu^k)' \cdot g \\
&\leq q(\mu^k) + (\mu - \mu^k)' \cdot g
\end{aligned} \tag{6.24}$$

The Lagrangian multipliers are updated by the following formula:

$$\mu^{k+1} = \left[\mu^k + s^k g^k \right]^+ \tag{6.25}$$

where s^k is a positive scalar step-size.

The step-size is determined so that:

$$0 < s^k < \frac{2 \left(q(\mu^*) - q(\mu^k) \right)}{\left\| g^k \right\|^2} \tag{6.26}$$

where $q(\mu^*)$ is the optimal value of (6.16) with μ^* being the solution. However, $q(\mu^*)$ is unknown and can be only estimated based on the result from the current iteration. To simplify the calculation of step-size, in this paper, we choose the step-size $s^k = 0.5/k$. Based on our experience, it provides quite fast convergence.

Once all Lagrange multipliers are updated at each node, link-state routing protocol will be employed to exchange μ_{ij} among nodes in the network. Based on this new information, each node again solves the problem (6.19) and (6.21) independently. The information about traffic originating from a source node passing through each lightpath is calculated at that node based on the shortest path routing found by Dijkstra. It is then sent to the source node of the lightpath. This information is neccessary to calculate total traffic passing through a lightpath and will be used to update the Lagrangian multipliers. The process is repeated until the dual function converges.

6.3.2 Heuristic adjustment

Due to integer variables, the dual problem does not provide strong duality to the primary one. Therefore, in the dual solution, some of the constraints, which are relaxed by the Lagrangian multipliers, might have been violated. So to construct a feasible solution, a heuristic algorithm has to be employed.

After a number of iterations, when the dual problem starts to converge, we observe that some lightpaths are always stably loaded and some have unstable load, which switches from high to low every iteration. Similarly, some traffic demands have a fixed route while others switch among several routes. This can be explained as follows. When the Lagrange multiplier of a lightpath is low, the lightpath will attract more traffic and consequently, that lightpath is overloaded. The Lagrange multiplier of that lightpath for the next iterration will be increased to avoid traffic passing through it. Hence, in the next iteration, the lightpath will be under-utilized, which in turn decreases the Lagrange multiplier for the next step. This phenomenon happens to some lightpaths in the network. Based on this observation, we propose some adjustments in the solution of the dual problem to generate the solution for the original problem.

6.3.2.1 Candidate lightpath adjustment

While solving the dual problem, every node considers the load of all lightpaths leaving it. If the load of a lightpath is stable for a long time and higher than the lower bound:

- If the lightpath is a candidate added lightpath, this lightpath is highly needed. We move this lightpath to the current topology and delete it from the candidate list.
- If the lightpath is a candidate deleted lightpath, we remove it from the candidate list and keep this lightpath as it is.

If a lightpath is always under-utilized:

- If the lightpath is a candidate added lightpath, remove it from the candidate list.
- Otherwise, remove it from the current topology y_{ij}.

Note that while doing this adjustment, the calculation of the dual function is still carried out. This adjustment will slowly fix all the neccessary lightpaths.

6.3.2.2 Traffic routing adjustment

As mentioned above, some node-pairs have traffic routed in a fixed route, while others have traffic switched among several routes. If a traffic demand switches among several routes, its source node will choose the shortest route (in terms of lightpath-hops). This step is done when the calculation of the dual function has been finished.

6.3.2.3 Final topology adjustment

After fixing the added/deleted lightpaths and traffic routing, each node simply considers all lightpaths outgoing from it. If any lightpath is overloaded, the node will request to create a new lightpath to the same destination node to share the traffic with the overloaded one. This step will eliminate the constraint violation.

6.3.3 Summary of the distributed algorithm

Once a node detects a violated lightpath, it broadcasts the reconfiguration-triggering message to all other nodes in the network. The link-state protocol is then used to exchange the information about traffic demands between node-pairs. This information is necessary to calculate candidate added/deleted lightpaths. Once all nodes have the complete current traffic pattern, they independently calculate the candidate lightpaths using the algorithm presented in Section 6.2.2. The routing and wavelength assignment for a candidate added lightpath can be done by the distributed algorithm presented in [RS03].

When the calculation of candidate lightpaths finishes, solving of the dual problem is started. Each node solves the problems (6.19) and (6.21), as well as does *candidate lightpath adjustment* separatedly and exchanges information with other nodes using link-state routing protocol. This process is repeated until the dual function converges. When the loop is terminated, each node does *traffic routing and final topology adjustment*. Once the new logical topology is successfully created, traffic will be routed onto it. Unused lightpaths are finally deleted. The reconfiguration then finishes.

Obviously, there is no central manager to calculate the dual function. We therefore do not know if the dual function converges and each node will not know when to stop the calculation process. However, based on experience, we can estimate the number of iterations for the dual function to converge. So at each node, the number of iterations can be set up in advance. This will be discussed more in the numerical example.

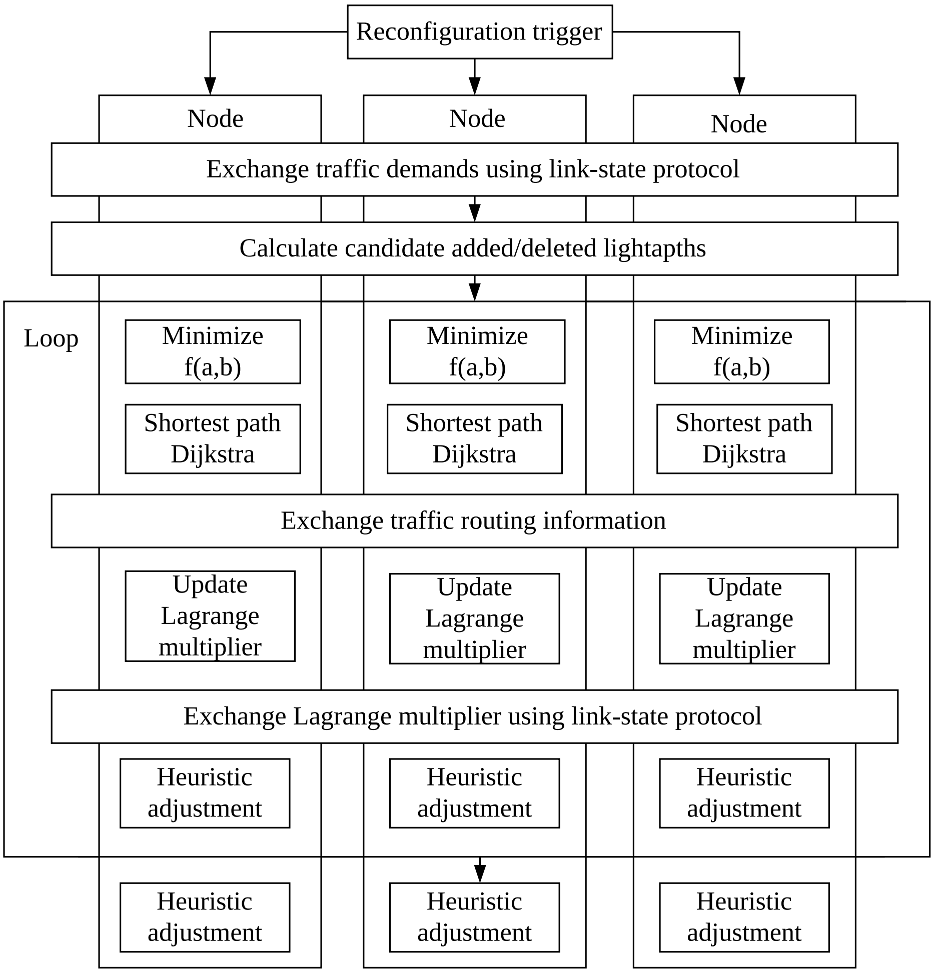


Figure 6.1: Diagram of the distributed algorithm

The procedure of the reconfiguration can be presented by the diagram in Fig (6.1)

6.4 Performance evaluation

6.4.1 Traffic model

In this study, we use traffic patterns taken from the GEANT project [UQBL06]. These data are measured in bytes every 15 minutes. They are pretty low compared to the bandwidth capacity of a wavelength channel. We therefore multiply all traffic demands by the factor of 15. This doesn't change the behaviour of the traffic fluctuation. In Fig (6.2), we see the variation of traffic of two node-pairs, as well as the total traffic in the whole network within one week (from Monday to Sunday). As can be seen, the traffic of a node-pair changes over time but does not have daily periodicity like the total traffic. That is why we cannot create just one logical topology, which can work for all traffic patterns. Dynamic reconfiguration of logical topologies is therefore neccessary for the IP over WDM network to work efficiently.

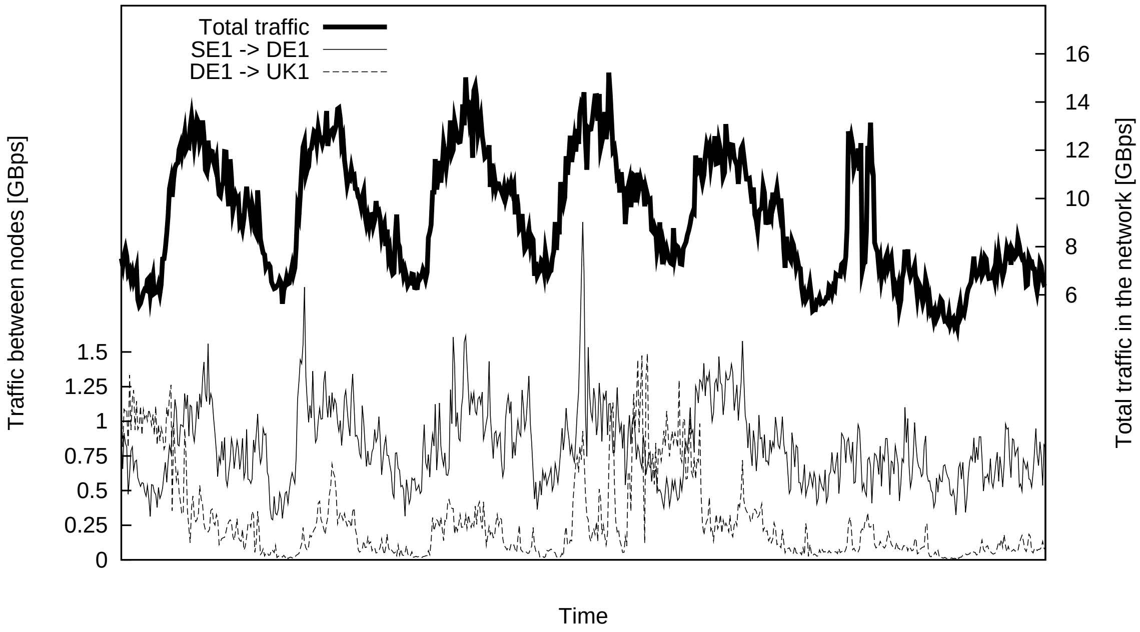


Figure 6.2: Traffic measurement from Geant network

6.4.2 Performance evaluation of the centralized approach

Since the ILP solution of the exact optimization problem is achievable only for small networks, we use the single-fiber 6- node, 7-link network in Fig. 6.3 as a numerical example for this logical topology reconfiguration problem. This topology is a subnet of GEANT network's topology (shown in Fig. 6.12). We assume that there are four transmitters and receivers at each node and six wavelengths multiplexed in a fiber. The wavelength capacity is 2.5Gbps. The upper bound and lower bound are chosen to be 80% and 10%, respectively. The *load balance indicator* L is set to 55%. It might happen that the optimization problem has no feasible solution due to the lack of free resources when the traffic is very high. In this case, L is increased 5% step by step until there is a feasible solution. While applying the approximate model, we assume that there are five candidate added lightpaths and five candidate deleted lightpaths for each reconfiguration. CPLEX is used to solve the optimization problem.

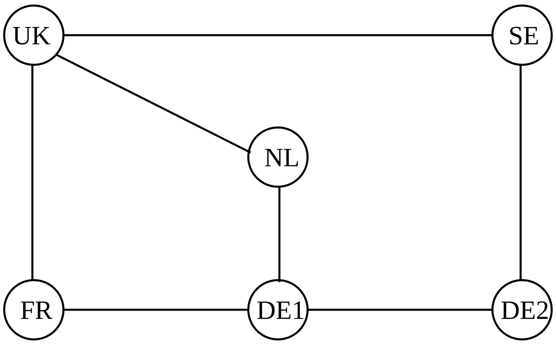


Figure 6.3: Six-node network physical topology

6.4.2.1 Comparision with Gencata's model

We first evaluate the performance of the centralized approaches by comparing it with Gencata's minimal lightpath change model.

The reconfiguration method proposed by Gencata et al in [GM03] can be summarized in Algorithm 3.

Algorithm 3 Gencata's minimal lightpath change model

```
measure lightpaths'load every $\Delta$ minutes
if some lightpath is highly loaded then
    add one lightpath so that the maximum lighpath load is minimized.
end if
if some lightpath is lightly loaded and no lightpath is highly loaded then
    delete one lightpath so that the maximum lighpath load is minimized.
end if
```

The change of the logical topology as a function of time for Gencata's and our approaches (both exact and approximate models) is shown in Fig. 6.4, Fig. 6.5, and Fig. 6.6, respectively. As can be seen, the number of lightpaths changes over time and also has daily periodicity like the traffic pattern in Fig. 6.2. The maximum and minimum lightpaths' load over time is also plotted.

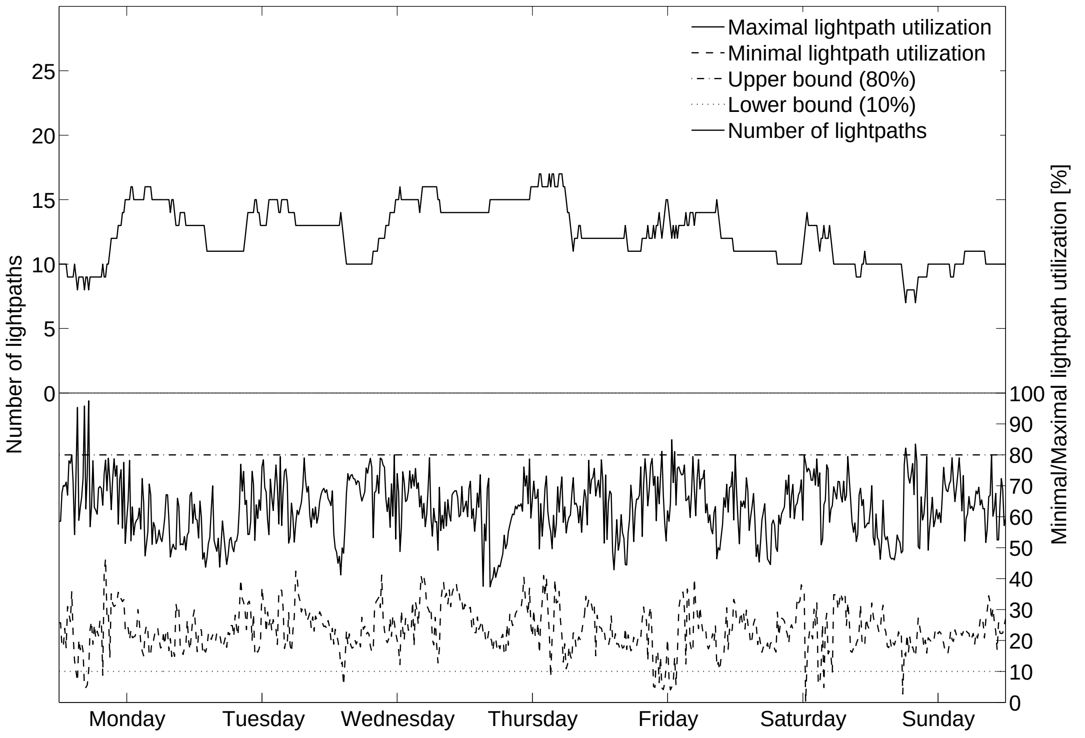


Figure 6.4: Time variation of logical topology, using Gencata model for reconfiguration

Gencata algorithm allows only one change at a time while ours allows to add and to delete multiple lightpaths. Moreover, in Gencatas algorithm, the decision of adding or deleting a lightpath is based on the lightpaths load. In our algorithm, lightpaths load is just a signal to trigger the reconfiguration process but the decision of adding or deleting lightpaths is the output of the optimization problem. It can happen that the number of deleted lightpaths is higher than that of added lightpaths even though some lightpath is highly loaded and vice-versa. Another feature of our algorithm is that in some case, we might not need to add or delete any lightpath but only do rerouting. This is, in our opinion, reasonable because a high loaded lightpath might be caused by the increase in traffic between only one node-pair but not by the total traffic in the network. Hence, it is

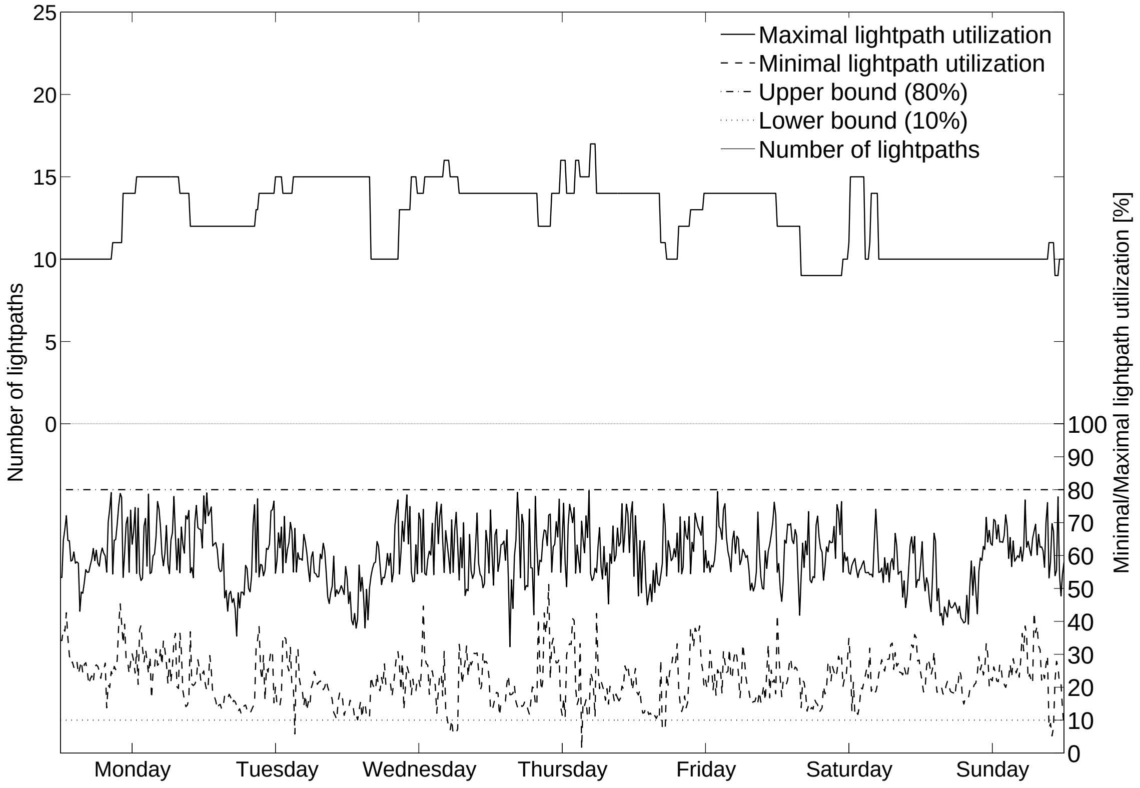


Figure 6.5: Time variation of logical topology, using our exact model for reconfiguration

not always necessary to add or delete lightpaths when some lightpaths are highly loaded or under utilized.

A statistical comparison between Gencata's and our algorithm is shown in Table 6.1. In the table, *real reconfiguration* refers to the reconfiguration in which some lighptpaths are added or deleted. This is distinguished from general reconfiguration where rerouting is also counted.

As can be seen, the number of reconfigurations in our both exact and approximate approaches is significantly lower than that of Gencata's even though we have to add/delete more lightpaths. Moreover, in our algorithm, the lightpaths's load is more balanced and there is no lightpath violating the upper bound whereas there are some in Gencata's approach. Violating the upper bound is very critical and should be avoided because it can cause traffic loss. This proves that changing only one lightpath at a time is not enough to guarantee network performance. For large networks with high number of lightpaths,

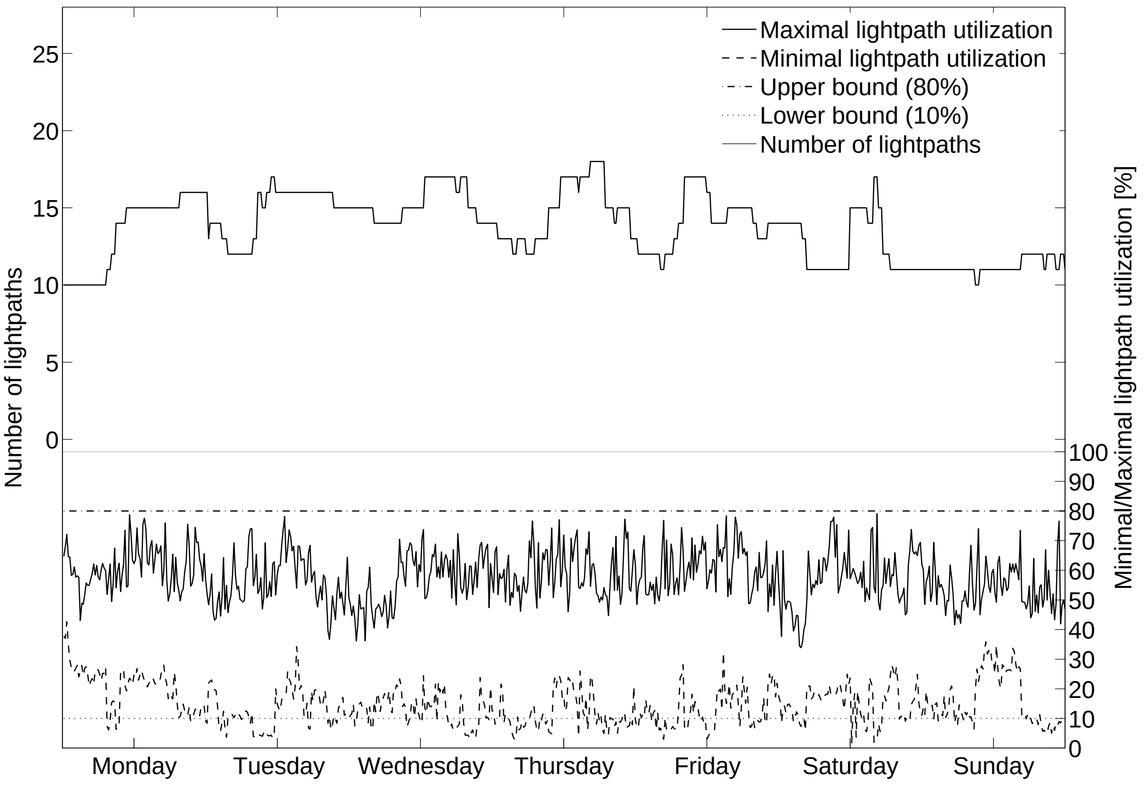


Figure 6.6: Logical topology reconfiguration using approximate model

Table 6.1: Comparison of Gencata's and our centralized approaches

	Exact model	Approximate model	Gencata's
The number of changed lightpaths	310	205	189
The number of reconfigurations	58	84	189
The number of real reconfigurations	57	83	189
Average reconfiguration period (min)	176	121	58
Violation of upper bound	0	0	0.31%
Violation of lower bound	0.21%	3.2%	2.9%

adding/deleting only one lightpath definitely cannot ensure a good performance.

Obviously, the exact model results in a better performance than the approximate model in terms of reconfiguration frequency and load balancing. The exact model has a lower number of reconfigurations as well as fewer lightpaths violating the balance region (0.21% vs. 3.2%). However, the number of changed lightpaths in the approximate model is much lower than that of the exact model (205 vs. 310, respectively). This is clear because the exact model searches for the new logical topology considering all lightpaths in the network, whereas the approximate model just looks for the new topology within a set of pre-determinded lightpaths, which is selected by a heuristic algorithm.

6.4.2.2 Effect of load balance indicator L

The load balance indicator L is an important parameter of the system. Different values of L can result in a different number of times the reconfiguration is triggered. To see the influence of L on the performance of the algorithm, we run the simulation using the exact model applied to the 6-node network for a period of one month and calculate the average number of reconfigurations in one day. The effect of L on the number of reconfigurations is shown in Fig. 6.7.

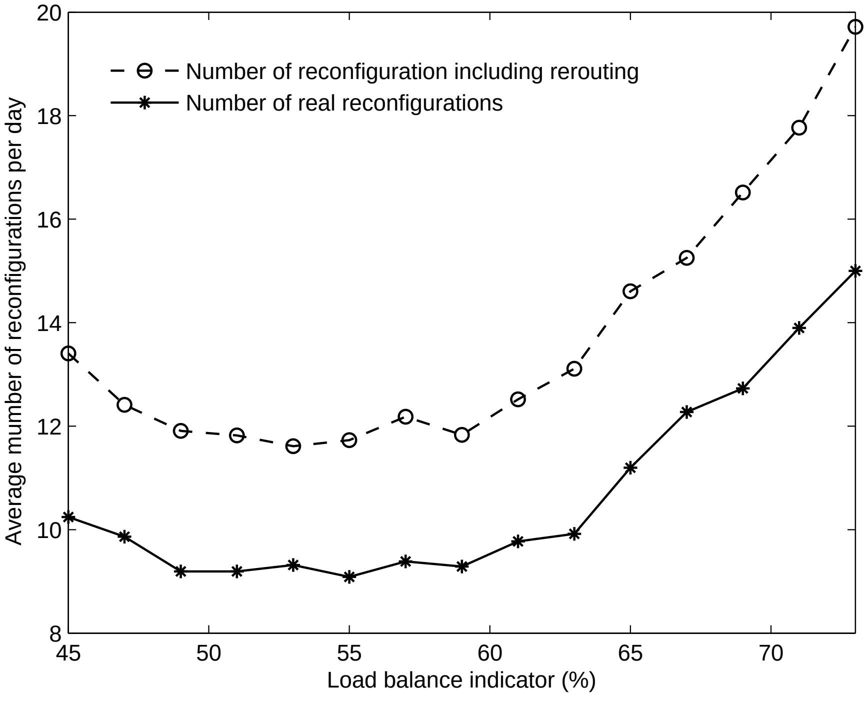


Figure 6.7: Influence of L on the number of reconfigurations

The figure above shows that a reasonable value of L is in the range from 53% to 62%, which results in around 11 reconfigurations per day, of which around 9 times include adding or deleting lightpaths while 2 times rerouting alone is done. The choice of the rerouting option is the strength of our algorithm. On the other hand, the ratio 2:9 shows that reconfiguration of the logical topology is a requirement for efficient usage of IP over WDM networks.

It is clear that if we choose a high value of L, the reconfiguration will be triggered more often because it is easy to violate the upper bound. In the other case, if L has a low value, the reconfiguration is also triggered more often because of violating the lower bound. So for a pair of (T_{high}, T_{low}), there exists an optimal value (or range) of L such that the number of reconfigurations for a long period is minimized. It is impossible to calculate the optimal L analytically because it depends on the traffic fluctuation as well as traffic distribution among node-pairs, which are unknown. However, with some assumption, we can estimate it roughly as follows.

According to the algorithm, after the reconfiguration, the highest lightpath load is L. Let $L - \Delta$ be the lowest lightpath load, with Δ being the difference between the highest and the lowest lightpath load. The total traffic goes up and down with a period of one day but the average traffic is stable for a period of months and we assume that the traffic fluctuations of node-pairs are independent from each other. We therefore can reasonably consider that at a given time, the probabilities of violating the upper bound and lower bound are equal. With this assumption, the distance between the highest lightpaths load and the upper bound should equal to the distance between the lowest lightpaths load and the lower bound. That is:

$$T_{high} - L = L - \Delta - T_{low} \tag{6.27}$$

and hence:

$$L = \frac{T_{high} + T_{low} + \Delta}{2} \tag{6.28}$$

$\Delta > 0$, so the lower bound of L is:

$$L > \frac{T_{high} + T_{low}}{2} \tag{6.29}$$

In our simulation, we observe that the difference between the maximal and minimal lightpath load after reconfiguration is mostly around 10% to 25%. So with $T_{high} = 80\%$

and $T_{low} = 10\%$, the best L will be from 50% to 58%, which conforms to the simulation results shown in Fig. 6.7. An improvement of the algorithm can be obtained by adapting L to changing trends of traffic.

Clearly, doing many reconfigurations in a short time is not desirable since it causes traffic fluctuation in the network and raises management cost. In our algorithm, we restrict the maximum lightpaths load to L at the time of reconfiguration and hence it is unlikely that two reconfigurations are triggered at two consecutively observations due to violating the upper bound. But similar events could happen with the violation of the low water mark. However, violating of the lower bound is not critical to the network performance and can be tolerated. We therefore can apply a so-called *softbound* approach to reduce the reconfiguration frequency. In the softbound approach, a reconfiguration is done only after three observations since the previous reconfiguration if in between there is no violation of the upper bound. This may result in more under-utilized lightpaths than the original approach, which can be referred to *hardbound* reconfiguration, but it significantly reduces the number of reconfigurations, which can be very costly. This is the trade off between load balancing and reconfiguration frequency.

6.4.3 Performance evaluation of distributed approach

6.4.3.1 Convergence of the dual problem

An important issue of the distributed algorithm is when the function converges so that the algorithm can be terminated. We first carry out an experiement for the 6-node network. Fig 6.8 and Fig 6.9 shows the variation of the Langrange multiplier of two specific links and Lagrange dual function during the reconfiguration process, respectively. As can be seen, the function converges after around 200 iterations. For a large network, we also get the same behavior. So in these numerical examples, we terminate the loop after 200 iterations. This parameter should be agreed at all nodes in advance to avoid any loss of synchronization. Experiments show that increasing the number of iterations doesn't increase the performance.

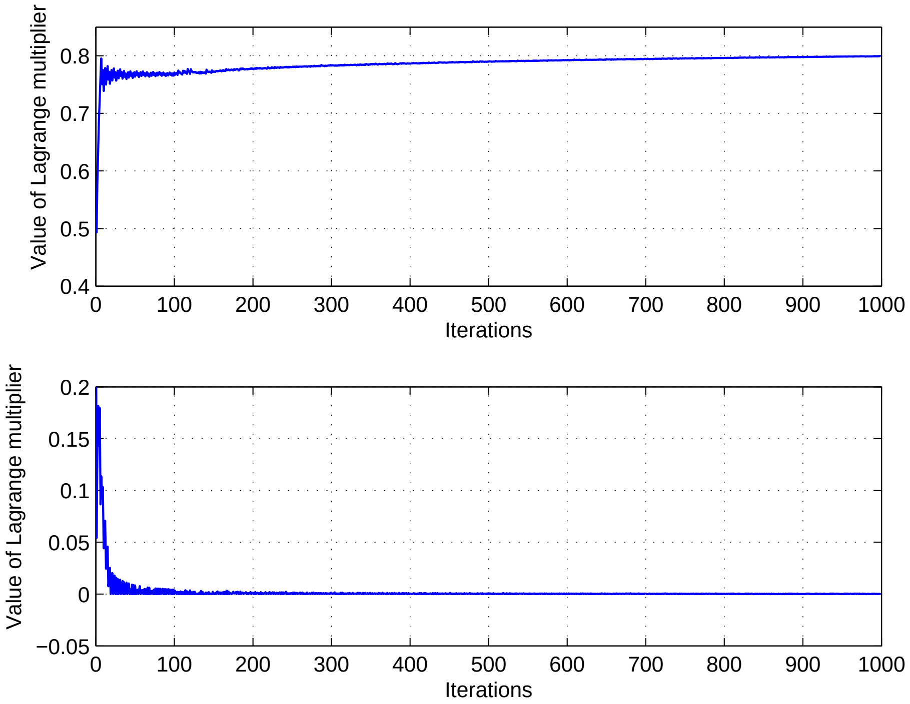


Figure 6.8: Lagrange multiplier changes during the reconfiguration process

6.4.3.2 Comparision with centralized approaches

We first compare the performance of the distributed algorithm with the centralized ones, using the 6-node network as shown in Fig. 6.3. The changing of the logical topology as a function of time for the distributed approach is shown in Fig. 6.10. Fig. 6.11 shows the lightpaths' load distribution for three models.

The main weakness of the distributed model compared to the centralized ones is the load balancing. From the lightpath load distribution, we can see that the number of lightpaths with high load ($> 60\%$) in the three models is more or less the same because the maximum load after the reconfiguration is limited to the same value of L. But the number of lightpaths with low load in the distributed model is clearly higher. Hence, more reconfigurations are triggered in the distributed model due to violation of the lower bound. To reduce the number of reconfigurations, we can apply the softbound approach presented in Section 6.4.2.2. Of course in this case, we have to accept a higher number of under-utilized lightpaths. We did a simulation, in which the reconfiguration is triggered only when the lower bound is violated three times continuously. As a result, only 88 reconfigurations are

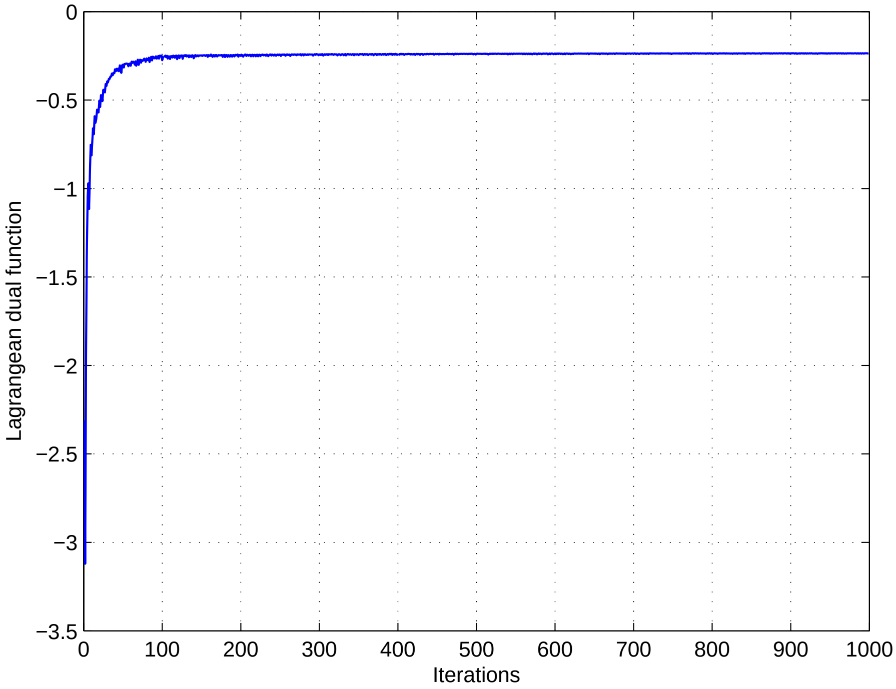


Figure 6.9: Lagrange dual function during the reconfiguration process

triggered. This is a good improvement while the number of lightpaths violating the lower bound doesn't increase much (5.6% in total).

A statistical comparison of three models is shown in Table 6.2.

Table 6.2: Comparison of centralized and distributed approaches

	Exact model	App. model	Dist. model
The number of changed lightpaths	310	205	368
The number of reconfigurations	58	84	119
The number of real reconfigurations	57	83	109
Average reconfiguration period (min)	176	121	92
Violation of upper bound	0	0	0.01%
Violation of lower bound	0.21%	3.2%	4.6%

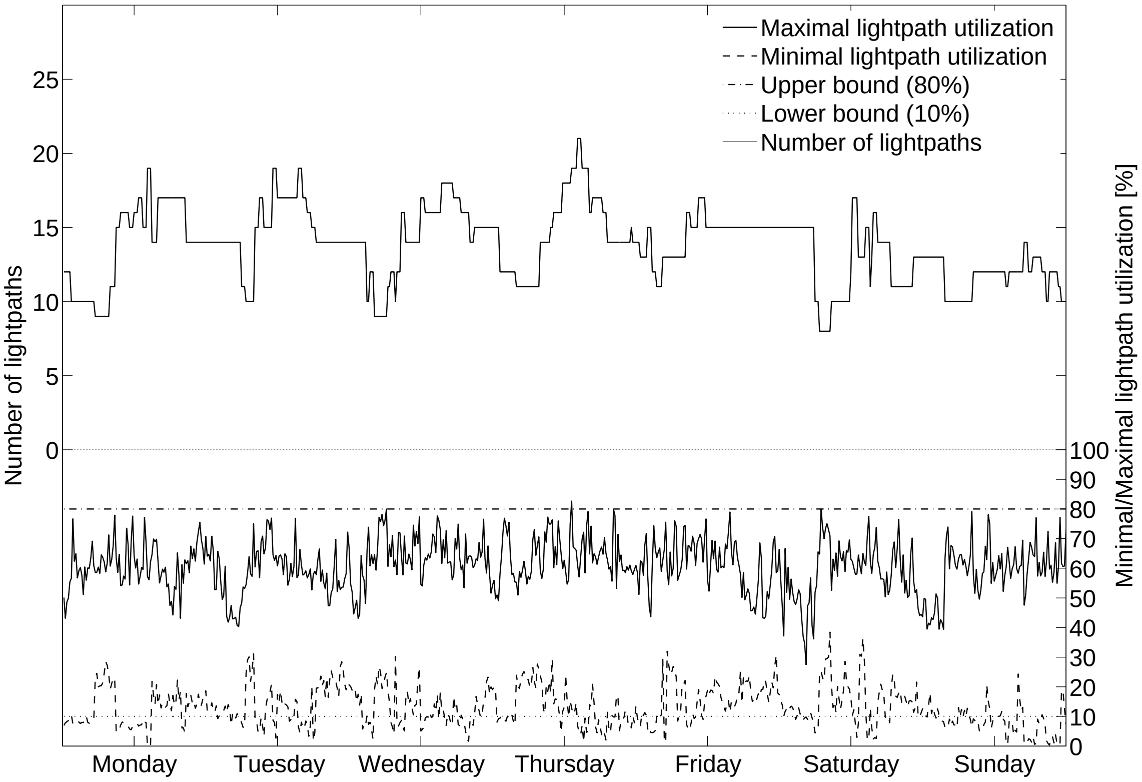


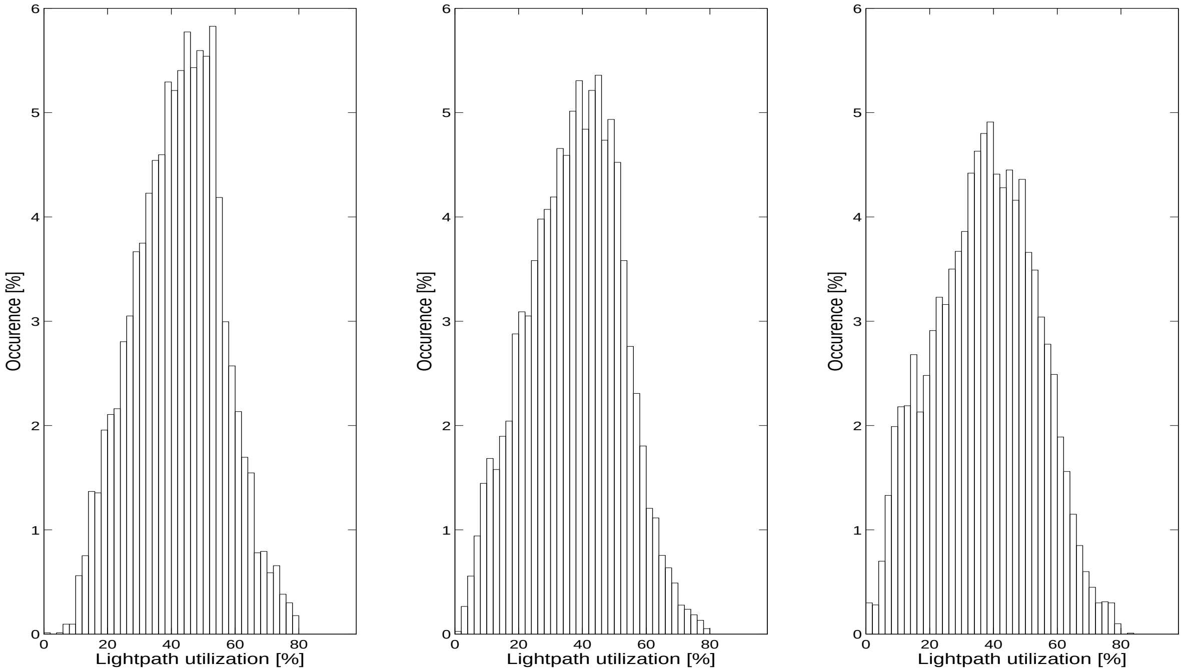
Figure 6.10: Logical topology reconfiguration using distributed algorithm

6.4.3.3 Numerical example with a large network

To confirm the scalability of the distributed algorithm, we carry out the simulation for 18-node, 27-link European network taken from GEANT project [UQBL06]. The network topology is shown in 6.12. We assume that each link is a single fiber with 16 multiplexed wavelengths and each node is equipped with 16 transceivers.

In this example, we discard the lower bound to eliminate the reconfiguration caused by violating the lower bound. In this network, traffic distribution among node-pairs differs a lot. A few nodes have a very low incoming and outgoing traffic (compared to the bandwidth of a lightpath, which is assumed to be 2.5Gbps), such as HR or SI. But we still have to set up lightpaths originating and terminating at these nodes to guarantee the network's connectivity. These lightpaths are hence always under-utilized and consequently, the reconfiguration is always triggered.

The simulation results of the distributed algorithm for 18-node network are shown in



a) b) c)

Figure 6.11: Lightpath load distribution for : a) exact model, b) approximate model, c) distributed model

Fig. 6.13. Fig. 6.14 shows the lightpaths' load distribution. Table 6.3 shows some statistical figures of merit.

For this network, the exact model cannot be applied. The approximate model takes several minutes to find the solution, which is quite long compared to the traffic measurement period of 15 minutes. Hence the approximate model cannot be implemented online. The distributed algorithm takes only few seconds to solve the problem. This is the main advantage of the distributed algorithm.

Table 6.3: Results of the distributed algorithm for 18-node network

Number of changed lightpaths	958
Number of reconfigurations	91
Average reconfiguration period	110
Violation of upper bound	0
% lightpaths with utilization ¡ 10%	9.6%

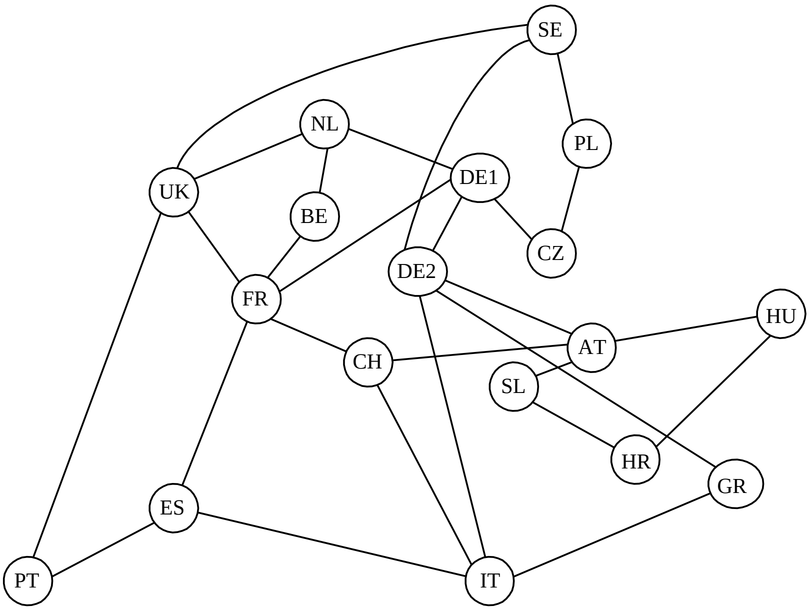


Figure 6.12: 18-node European network

6.4.3.4 Computation time of the distributed algorithm

The computation at each node is quite simple. However, the whole reconfiguration process requires a lot of message exchanges. We are therefore interested in the computation time of a reconfiguration.

Following the diagram in Fig 6.1, we see that the computation time includes the time to exchange traffic demands, to calculate candidate lightpaths, to solve dual problems and to do the heuristic adjustment. However, it is obvious that the time to solve dual problems dominates the others because it requires hundreds of iterations. So we can consider the computation time equal to the time to fulfil the loop. It can be calculated as follows (with the assumption that ACKs are needed for every sent message):

$$T = I \cdot (2P + S)$$

where T is the computation time, I is the number of iterations, P is the propagation delay for a message and S is the time to solve the problems (6.19) and (6.21) at each node. The transmission delay is discarded because it is too small compared to the propagation delay. We assume:

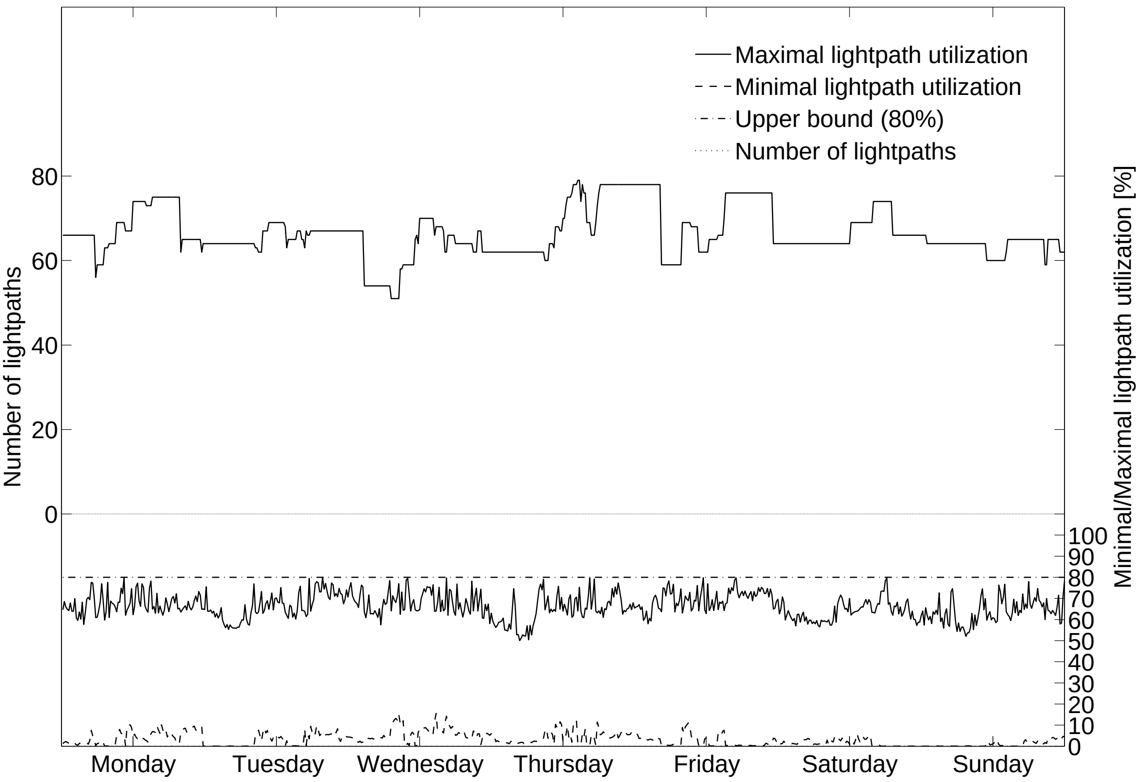


Figure 6.13: Logical topology reconfiguration for 18-node network

- The average propagation delay on each physical link is 2.5 ms, corresponding to a link length of 500km. Some nodes have a shortest path of 5 physical links. So the message propagation delay is about 12.5ms.
- The time to solve the problems (6.19) and (6.21) for each node is roughly 5ms using 2GHz CPU (according to our experiments).
- The number of iterations is 200.

We have:

$$T = 200(2 \cdot 12.5 + 5)\mu s = 6s$$

So roughly, the computation time for a reconfiguration is 6s. This is clearly much smaller than the measurement period, which is 15 minutes. This is even smaller than the computation time of the exact model for small networks, which is usually arround 30s. The distributed algorithm hence can be implemented online.

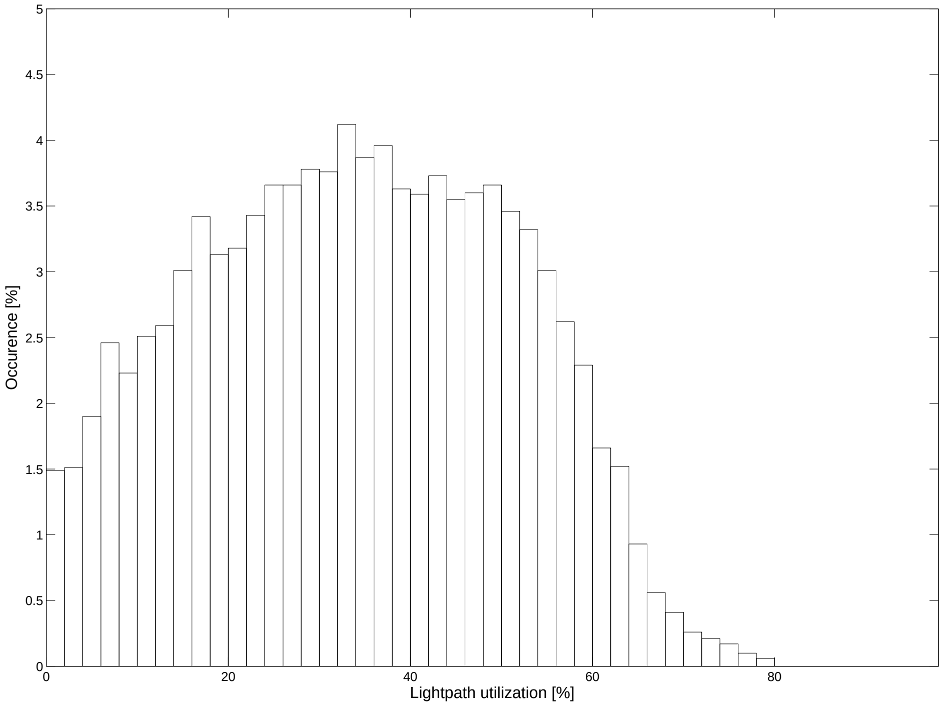


Figure 6.14: Lightpath utilization distribution

6.5 Conclusion

In this chapter, we studied the problem of logical topology reconfiguration for IP over WDM networks. Since the traffic changes over time, the logical topology needs to be adapted accordingly to efficiently carry the traffic. In this work, two approaches, i.e centralized and distributed approaches were proposed to solve the problem. In the centralized approach, two mathematical models using ILP formulation were developed. The first model solved the problem in an exact manner. The approximate model simplifies the exact model by removing the RWA constraints for new lightpaths to reduce the complexity. In this approach, the reconfiguration problem was solved by a central manager, who was assumed to have the global information about network resources as well as traffic loads.

Based on the approximate model, a distributed algorithm was developed using the Lagrangian relaxation method. The capacity constraint was first moved to the objective function to form a dual problem. This dual problem could be divided into independant subproblems and hence could be solved in a distributed manner by all nodes in the network. Link-state routing protocol was used to exchange messages among the nodes. Even though the performance degrades in terms of reconfiguration frequency, the distributed algorithm still satisfies the requirements of a reconfiguration. The main advantage of the distributed algorithm compared to the centralized approaches is its scalability. In addition, this distributed algorithm can lend itself to a protocol implementation.

Chapter 7

Survivability in two-layer WDM networks

This chapter discusses the survivability of two-layer WDM networks, using IP/MPLS over WDM as a network reference. Like all networks, WDM networks are prone to failures of components such as links and nodes. But different from other conventional copper networks, survivability is a very critical problem in WDM networks. Since these networks carry high volumes of traffic, just a single optical link failure may cause severe consequences. Therefore, an efficient protection mechanism is required to guarantee the Quality of Service in WDM networks.

For static traffic, protection and restoration define the survivable network design problem, which can be formulated as an ILP problem. The objective of this problem is often to minimize the required sparse resources whereas ensuring that all traffic demands have a backup path that cannot fail at the same time with the primary path. In this thesis, we won't consider the static traffic case but just the dynamic traffic case, where LSP (Label-Switched Path) requests come and leave the network in a random manner. In a dynamic traffic environment, the objective is to increase the acceptance ratio (or to decrease the blocking probability equivalently). Protection for dynamic traffic can be handled only by a heuristic algorithm.

Conventionally, the protection mechanism for dynamic traffic works as follows. When a connection request arrives, two link-disjoint (or node-disjoint in case of node failure protection) paths are assigned to the connection. One acts as a primary path, which carries traffic in normal case and the other is the backup path used in case a failure occurs. These two paths will be realeased when the connection terminates. The backup path can be either a dedicated path, which protects only one primary path or a shared backup path, which protects several primary paths that cannot fail at the same time. The dedicated protection mechanism results in short restoration time whilst the shared protection mechanism results in a better resource utilization. Different protection mechanisms have different strategies to select the primary path and backup path. In multi-layer networks, the protection mechanism also works as the before-mentioned mechanism. However, the challenge of multi-layer protection lies in the fact that link-disjoint (or node-disjoint) paths in the upper layer may not be link-disjoint (or node-disjoint) in the lower layer. Hence, searching for disjoint paths becomes more difficult.

In this thesis, we propose a new protection mechanism for IP/MPLS over WDM networks. Different from all conventional approaches, which assign backup paths on demand together with the arrival of new connection requests, our approach first creates a so-called protection topology in the logical layer. This protection topology is designed offline in such a way that all nodes remain connected when a single optical fiber fails. So basically, any connection between two nodes can be protected by a path in this protection topology. An approriate amount of bandwith will be reserved in this topology for protection purpose. When a connection request comes, one has to assign only a primary path to it, which significantly reduces the computation overheads.

In the scope of this work, we only consider the single link failure model. This model presumes that at most one link has failed at any instant of time. Since 99% of failure happens at only one link at a time, this model covers almost all cases in link-failure protection and restoration.

7.1 Protection mechanism

7.1.1 Network model

In this thesis, we use IP/MPLS over WDM as a network reference to study multilayer survivability problem. MPLS classifies packets arriving at the edge routers into forwarding equivalence classes and then forwards the packets with labels along label switched paths (LSPs). This enhances the quality of service (QoS), which makes IP/MPLS over WDM be a potential candidate for Next Generation Internet core networks.

In IP/MPLS over WDM networks, OXCs are inter-connected by fiber links whilst Label Switched Routers (LSRs) are connected to the OXCs through wavelength ports. An LSR has opto-electronic conversion and sub-wavelength multiplexing capabilites to aggregate multiple LSPs into lightpaths that can be dynamically provisioned. These lightpaths define a logical topology for the IP/MPLS layer. The optical layer manages the the resources including wavelengths and fibers on physical links whereas the IP/MPLS layer manages bandwidth resources on lightpaths and routes traffic over the logical topology considering a lightpath as a link. With the capabilities of MPLS such as QoS provisioning and traffic engineering, such networks enable direct integration of IP and WDM without intermediate layer ATM or SONET.

IP/MPLS over WDM networks may use either an overlay model or a peer model. In the overlay model, the IP/MPLS layer and optical layer are controlled separately. Each layer has its own control plane. In the peer model, the optical and IP/MPLS control planes are unified to provide efficient management and usage of network resources. The topology perceived by the network nodes is the one where physical links (optical fibers) and logical links (lightpaths) co-exist [KL01]. Different from the overlay model, the peer model supports dynamic routing that can either use existing lightpaths or create a new lightpath if neccessary.

7.1.2 Overview of the proposed protection mechanism

For a given IP/MPLS over WDM network, our protection scheme starts with creating a protection topology in IP/MPLS layer. This topology is designed in such a way that it connects all nodes in the network and maintains connectivity under any single-link failure. So the backup path of a connection between any two nodes is determined in advance. This topology should use as few wavelength resources as possible to increase the resource efficiency. The design of this protection topology can be formulated as an ILP problem, which will be discussed in the next section in detail. An amount of wavelength resources will be reserved for this topology to protect LSP requests. When an LSP comes, one has to assign only a primary path to it and reserve a sufficient bandwidth for its backup path in the protection topology. LSPs that do not share the same physical link can share the same backup bandwidth to increase the resource efficiency.

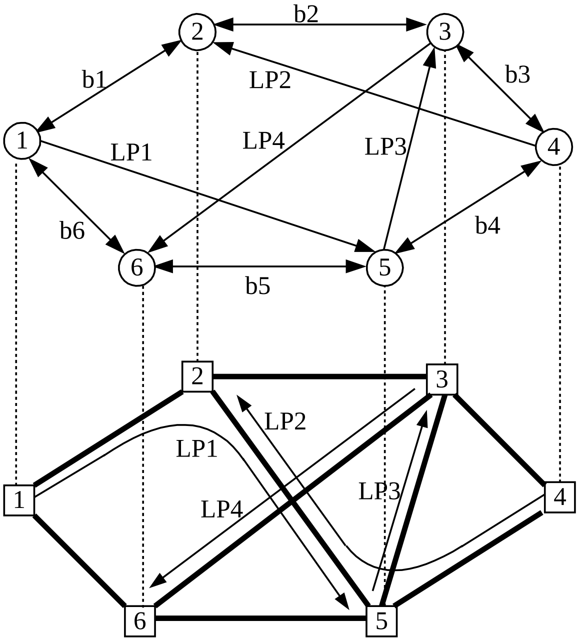


Figure 7.1: Example of pre-created protection topology

An illustrative example of the protection mechanism is depicted in Fig. 7.1. The protection topology is the cycle composed by a set of lightpaths ($b_1 - b_6$), which goes through all nodes in the network. This topology always remains connected under any single-link failure. A set of four lightpaths ($LP1, LP2, LP3, LP4$) forms a logical topology to carry LSP requests. Let's assume that there is an LSP request from node 1 to node 3. This LSP will be routed through two lightpaths $LP1$ and $LP3$. The potential break of this LSP is due to the failure of the physical links $(1-2)$, $(2-5)$ or $(5-3)$. In any case, node 1 and 3 can be connected through the backup lightpaths (b_6, b_5, b_4, b_3). So we just need to researve a sufficient bandwidth along this path to protect the LSP. Assume that another LSP request from node 3 to node 6 comes. This LSP will be routed through the lightpath $LP4$ and be protected by the path (b_5, b_4, b_3). These two LSPs cannot fail at the same time because they don't traverse the same physical links. Thus, they can share the backup bandwidth in the protection topology. This mechanism behaves similar to *p-cycle* protection in single-layer networks. *p-cycle* is actually the inspiration for us to propose this protection mechanism for multi-layer networks.

7.1.3 ILP formulation for Protection topology design

In this section, we introduce the ILP formulation for the design of the protection cycle. The topology is designed in such a way that when a physical link fails, the IP network is still connected and the necessary wavelength resources used for the topology are minimized. The following notation is used in the formulation:

- $L = \{l\}$ denotes the set of physical links.
- s and d denote the source and destination nodes of a LSP, respectively.
- i and j denote the originating and terminating nodes of a lightpath, respectively.
- $R = \{r\}$ denotes the set of alternative routes for a lightpath at the physical layer.

Given parameters

- $\delta_{r,l}^{ij}$ denotes the existence of physical link l on route r of lightpath (i, j).

Decision variables

- x_{ij}^{l}: a binary variable, which represents the existence of a lightpath between between i and j in case link l fails. $l = \varnothing$ denotes no failure . $x_{ij}^{l} = x_{ji}^{l}$.
- $t_{ij}^{sd,l}$: flow of node-pair (s, d) going through lightpath (i, j) when link l fails. $t_{ij}^{sd,l} = t_{ji}^{ds,l}$
- p_r^{ij}: denotes the r-th route used for lightpath connecting node i and j.

Constraints

$$\sum_j t_{ij}^{sd,l} - \sum_j t_{ji}^{sd,l} = \begin{cases} 1 & s = i \\ -1 & d = i \\ 0 & s \neq i, d \neq i \end{cases} \qquad , \forall s, d \tag{7.1}$$

$$t_{ij}^{sd,l} \leq x_{ij}^{l} \qquad , \forall i, j, l \tag{7.2}$$

$$x_{ij}^{\varnothing} = \sum_{r} p_r^{ij} \qquad ,\forall i, j \tag{7.3}$$

$$x_{ij}^{l} = \sum_{r} (1 - \delta_{r,l}^{ij}) \cdot p_r^{ij} \qquad ,\forall i, j, l \tag{7.4}$$

Equation 7.1 is a multi-commodity flow equation representing the routing of a LSP between (s, d) on lightpath (i, j). Constraint 7.2 ensures that LSP can only traverse an existing lightpath. Constraint 7.1 and 7.2 actually guarantee the connectivity of the logical topology in case of any failure. Constraint 7.3 represents the routing of a lightpath in WDM layer. Constraint 7.4 shows that lightpath (i, j) will break when link l fails in case link l belongs to the chosen route r of lightpath (i, j).

Objective

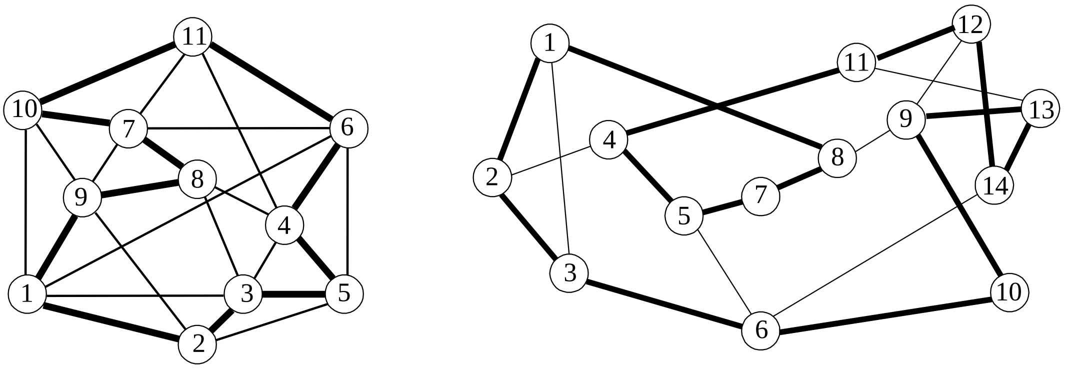
$$\min \sum_{ij} \sum_{r} \sum_{l} \delta_{r,l}^{ij} \cdot p_r^{ij} \tag{7.5}$$

The objective is to minimize the number of optical links used for the protection topology. Clearly, if resources are efficiently used, the blocking probability will decrease.

7.2 Performance evaluation

This section investigate the performance of the proposed multi-layer protection mechanism, using two network references: COST239 [COS] 11-node and NSFNET 14-node network as shown in Fig. 7.2. The protection topologies for these networks are depicted in bold lines. The protection topology is in fact a cycle that visits each node exactly once. Obviously, this is the connected topology with the lowest wavelength resource consumption possible. We assume that there are 16 wavelengths multiplexed in an optical fiber and 8 wavelengths are reserved for the protection topology.

Arrivals and holding time of LSP requests are assumed to follow a Poisson distribution and negative exponential distribution, respectively. The source and destination of an LSP is uniformly distributed in the network. We assume that a lightpath has a capacity of 10 and the request bandwidth of an LSP is uniformly distributed in the range from 1 to 10.



a) COST239

b) NSFNET

Figure 7.2: Investigated networks with protection topology

In this work, we use the hop-based integrated routing algorithm (HIRA) [ZM03a] to create lightpaths and assign a route to LSP requests. In this integrated routing algorithm, an LSP can traverse some existing lightpaths and some physical links with free wavelength resources to open new lightpaths. The objective of HIRA is to minimize the total number of physical hops used by the existing lightpaths and new lightpaths created for routing LSP requests. By doing so, it attempts to minimize the used wavelength resources, which will possibly lead to increased acceptance rate of future requests. In addition, it also likely reduces the number of opto-electronic conversions of LSPs.

7.2.1 Blocking probability

We first compare our proposed protection mechanism with two approaches, i.e WDM layer protection and IP layer protection. In general, IP layer protection results in a lower blocking probability than WDM layer protection. The hybrid protection schemes introduced in [RMZ05, ZM03b, RZG06] result in a blocking probability in between these two approaches.

Fig. 7.3 and Fig. 7.4 show the performance of the proposed mechanism applied to COST239 network and NSFNET network, respectively.

In both figures, we see that at high traffic load, the proposed protection mechanism performs very well. In COST239 network, the blocking probability is even lower than that in IP layer protection, which is the lower bound for all existing hybrid multi-layer

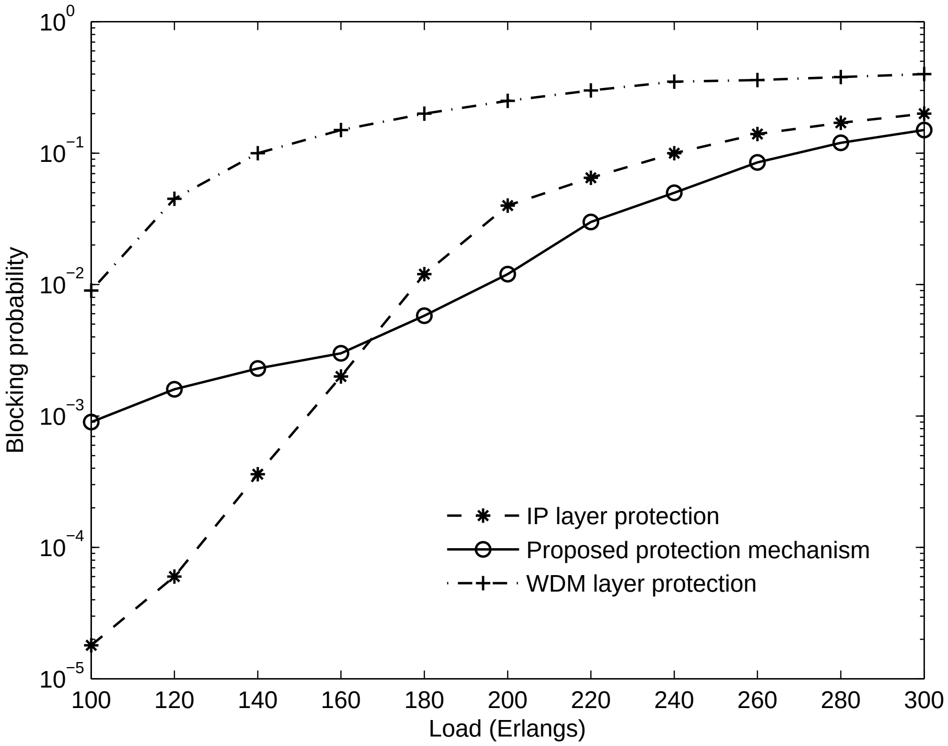


Figure 7.3: Blocking probability for COST239 network

protection mechanisms. However, at low traffic load, the proposed mechanism performs worse and worse. To overcome this phenomenon, we examined the reasons for blocking. As we observed in the experiements, blocking in our proposed mechanism occurs because of the following reasons:

- There is not enough resource for a new LSP request.
- There is not enough resource in the protection topology to provide a backup path for a new LSP request.

According to our statistics, when the traffic load is low, the blocking mostly occurs due to the lack of resource in the protection topology while at high traffic load, the blocking often happens because there is no resource for the primary path of a new LSP request. Therefore, it makes sense to adapt the reserved resource for protection purpose to the traffic load. We thus increase the number of wavelengths reserved for protection topology from eight to ten wavelengths and re-run the simulation for the region where the proposed

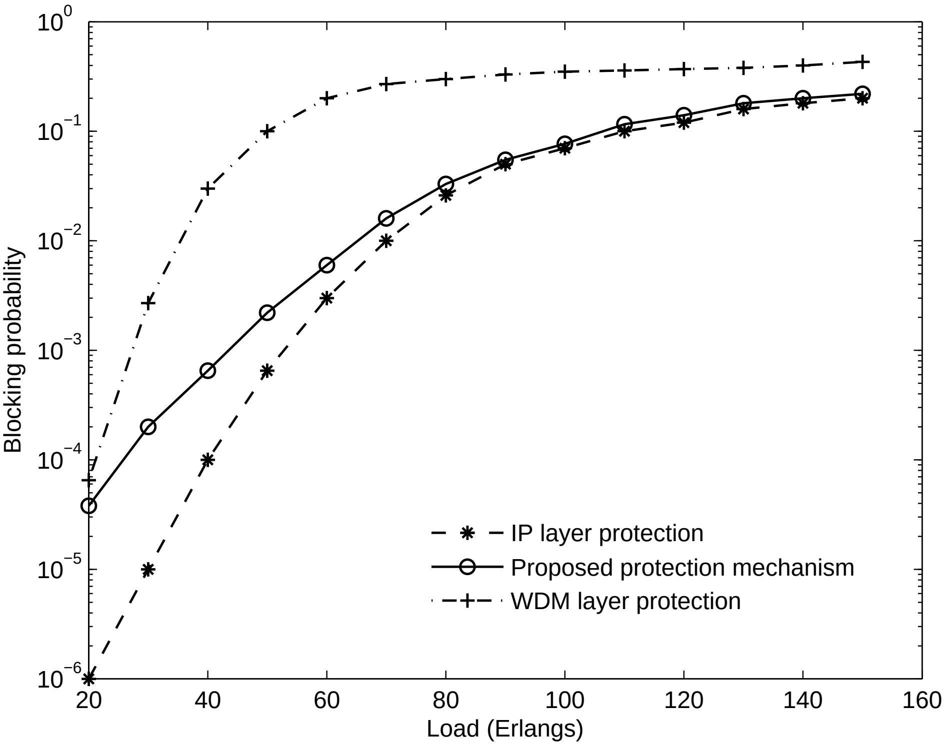


Figure 7.4: Blocking probability for NSFNET network

mechanism gets worse. The results are shown in Fig. 7.5 and Fig. 7.6 for COST239 and NSFNET networks, respectively.

From both figures, we see that the performance is improved significantly. The blocking probability of the proposed mechanism for both networks is close to that of the IP layer protection mechanism. So to summarize, our approach results in a very good performance in terms of blocking probability in comparison with both conventional protection approaches (WDM layer protection and IP layer protection) if an appropriate ammount of wavelengths are researved for the protection topology. By monitoring and measuring the traffic load in the network, one can actively add or release reserved capacities for the protection topology to get a better performance.

The reason that the protection mechanism performs slightly better in COST239 than in NSFNET is because COST239 network has a higher degree of interconnection. The protection topology in COST239 network can protect more LSPs than in NSFNET.

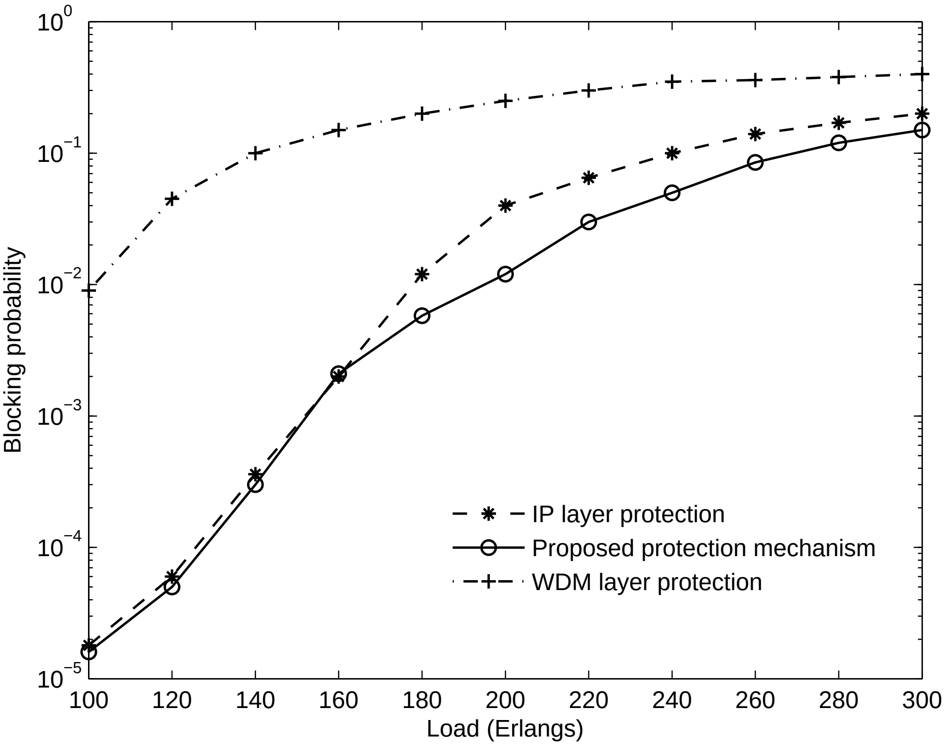


Figure 7.5: Blocking probability for COST239 network with varied researved bandwidth for protection

7.2.2 Analysis of backup paths' length

The obvious disadvantage of the proposed protection approach is the length (in terms of hop counts in the logical layer) of back-up paths. To minimize the wavelength resources used for the protection purpose, the protection topology tends to become a cycle that visits all N nodes in the network exactly once using the shortest physical path. So in the worst case, a backup path can have the length of $N-1$ hops. As can be seen in Fig 7.2a, in the worst case, a backup LSP may have to traverse up to 10 ligthpaths for the backup of a connection between nodes 1 and 2 e.g in case the link $(1-2)$ fails. To have a clearer view of this issue, we consider statistics of the distribution of backup paths's length.

Fig. 7.7 shows the comparison of the backup paths' length (in terms of hop counts in logical layer) in IP layer protection mechanism and our proposed one for COST239 network. In IP layer protection, most of backup paths have the length of two and maximum three logical links while in our protection mechanism, the paths' length ranges from one to ten

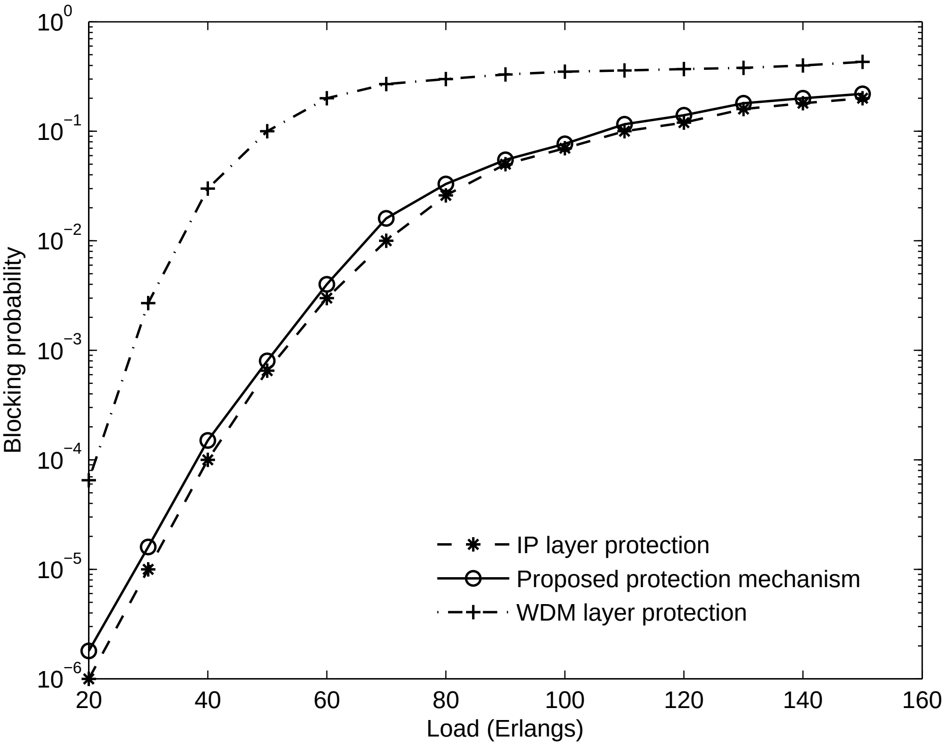


Figure 7.6: Blocking probability for NSFNET network with varied researved bandwidth for protection

logical links.

However, this disadvantage can be overcome by restricting the length of backup paths. A constraint, which limits the number of logical hops, can be added to the ILP formulation presented in Section 7.1.2 to reduce the length of backup paths.

$$\sum_{ij} t_{ij}^{sd,l} \leq H \qquad ,\forall sd, l \tag{7.6}$$

where H is the number of maximum hops allowed for a backup path in the logical layer.

Fig. 7.8 shows the protection topologies for COST239 network with the length of backup-paths restricted to 8 and 6 logical hops. Clearly, to reduce the hop-based length, more lightpaths need to be created, and thus more wavelength resources are needed for the protection topology. Compared to the case in Fig. 7.2a, limiting backup path length to eight or six logical hops requires two or one more physical links, respectively. We hence

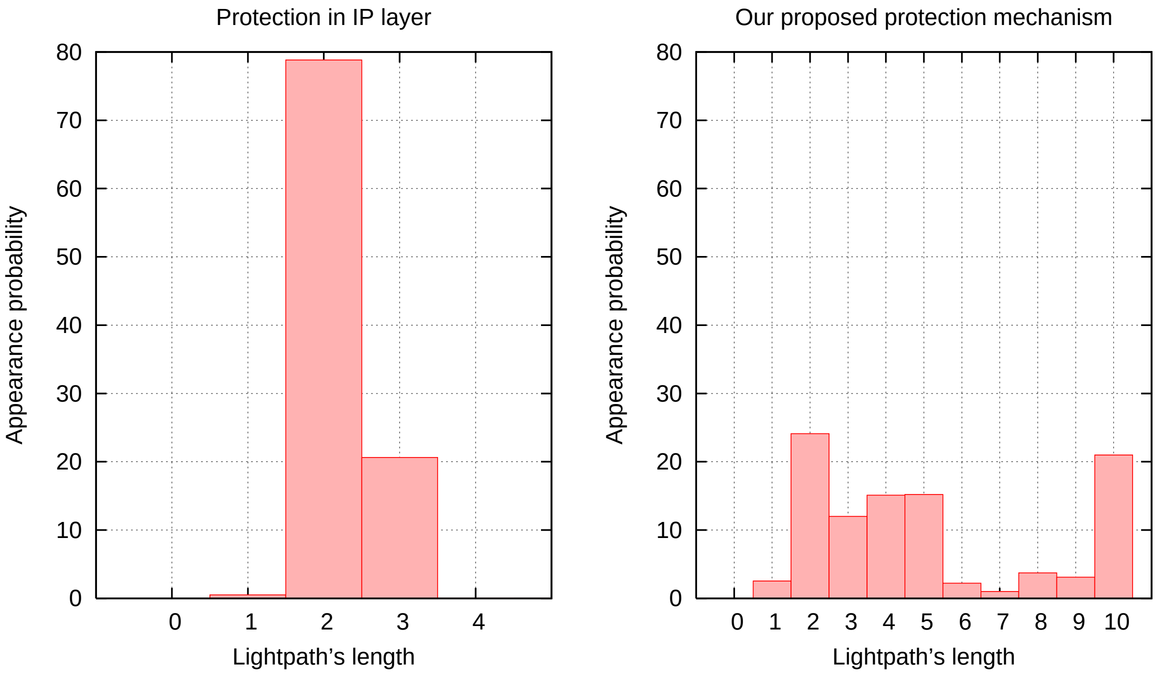


Figure 7.7: Comparison in the protection path length

want to see if there is any increase in blocking probability while using these two protection topologies.

Fig. 7.9 shows the blocking probability of COST239 network using three different protection topologies. In this simulation, from eight to ten wavelengths are researved for the protection topologies depending on the traffic load in the network. It is surprising that there is no significant difference among the three protection cases. This can be explained as follows. From Fig. 7.7, we see that there are aboubt 20% of backup paths have the length of 10 logical hops, which leads to much bandwidth needed to protect an LSP. When the protection path length is restricted, the total bandwidth (sum of bandwidth in all logical links) needed to protect an LSP is reduced, and hence the protection topology can protect more LSPs. Consequently, the blocking probability remains more or less the same in all three cases even though more wavelength resources are reserved for the protection topology.

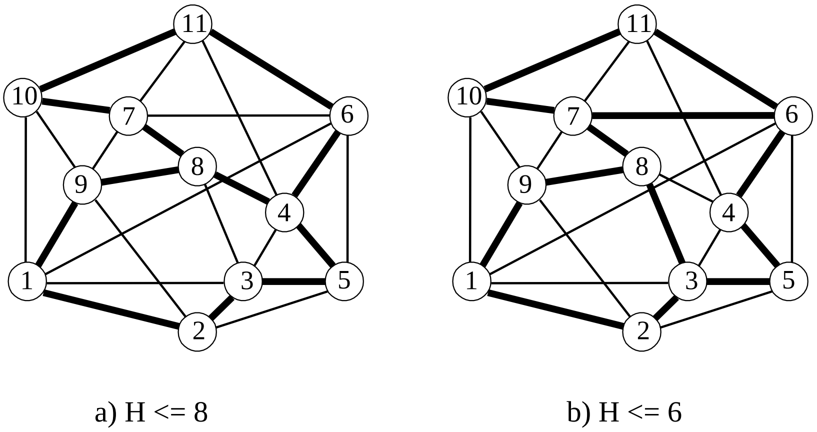


Figure 7.8: COST239 network with different protection topologies

Fig. 7.10 shows the comparison in backup paths' length of three protection schemes. As expected, the average length of backup paths reduces when we decrease the maximum protection path length allowed. The only trade-off while reducing the backup path length is the restoration time. As an example in Fig. 7.8a, the backup path that crosses node 4 and 8 is not switched in advance at these nodes. This can increase the recovery time. The shorter is the average protection path length, the longer is the average recovery time.

7.2.3 Significances of the proposed protection mechanism

After evaluating the performance of the proposed protection mechanism, we summarize its significance compared to the existing approaches as follows.

- The blocking probability is low. It is close to or even lower than that of the IP layer protection scheme if the network is highly meshed and an appropriate amount of wavelength resources is reserved for the protection topology.
- Even though this is a shared path protection strategy, the restoration time is short because all backup paths are created and switched in advance. In addition, it requires much less signalling overhead compared to the IP layer protection.
- Setting up an LSP request is more simple and faster than all existing approaches because one needs to find only a primary path for the LSP. In other approaches, one needs to find two link-disjoint paths, which can be computationally complex.

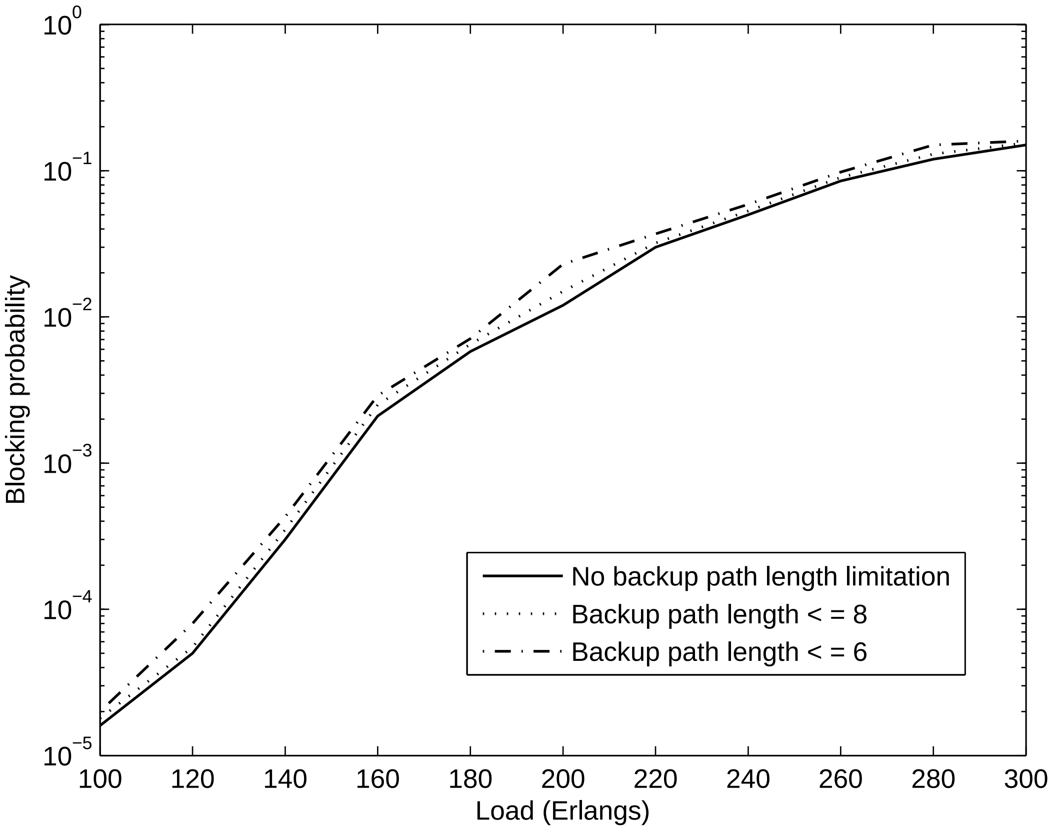


Figure 7.9: Blocking probability of COST239 network with different protection topologies

- Different from the existing hybrid multilayer protection schemes, the proposed mechanism doesn't require to monitor traffic or network status to decide on the protection strategy. This avoids additional signalling overhead to operate the network.

7.3 Conclusion

In this chapter, we proposed a new mechanism to protect IP/MPLS over WDM networks from a single link failure. Different from all existing approaches, which assign a back-up path for an LSP request on demand, our approach reserves the resource for protection purpose all in advance by creating a protection topology. The protection topology was designed in such a way that any single link failure in the physical layer does not disconnect the protection topology. Thus a backup path for a connection between any two nodes can

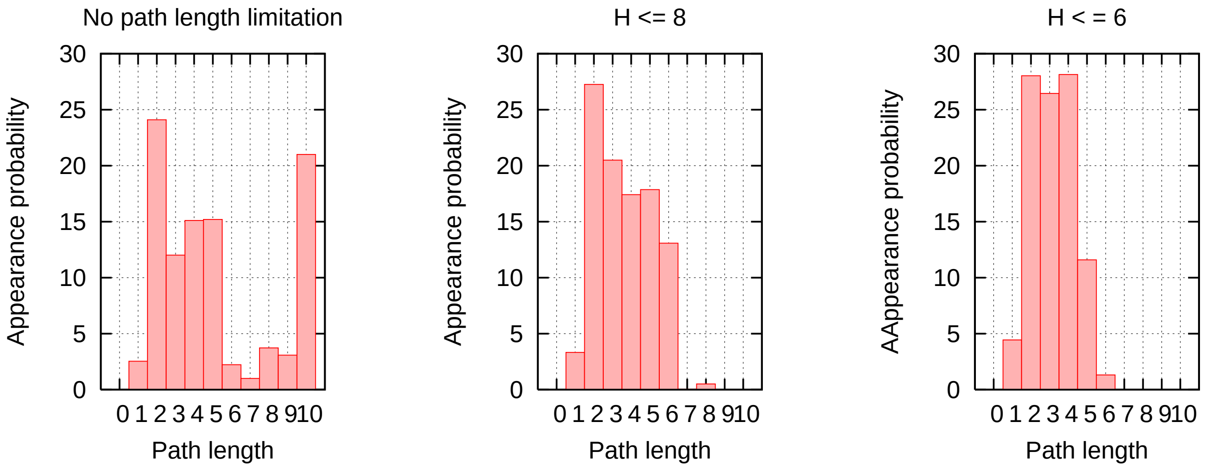


Figure 7.10: Comparison in protection paths' length of three protection schemes

be pre-determined beforehand. This helps to reduce the setup time for an LSP request because one does not have to find two link-disjoint paths for a request, which is not trivial.

If a proper amount of wavelength resources is reserved for the protection topology, the performance of the proposed mechanism (in terms of blocking probability) can be very close to or even better than that of the IP layer protection. Moreover, a much lower amount of signalling overhead compared to the IP layer protection is needed when failure occurs. Long protection paths can be avoided by limitting the maximum hop-count in the logical layer of backup paths to a certain value. This does not degrade the performance in terms of blocking probability much but leads to higher average restoration time.

Chapter 8

Conclusions

8.1 Summary

The rapid growth of Internet as well as bandwidth-intensive networking applications such as video-on-demand, multimedia conferencing, online game, etc. requires extensive new research in high-bandwidth transport networks, of which optical networks employing WDM technology is a promising candidate. In this thesis, we studied several issues related to wavelength routed networks, the potential candidate for wide area backbone networks. It included i) network planning and optimization for WDM networks, ii) wavelength converter placement and wavelength assignment in wavelength converting networks, iii) design and iv) reconfiguration of logical topologies for IP over WDM networks, and v) survivability for multi-layer WDM networks.

Network planning and optimization for WDM networks was first discussed since this was the base method we used to solve further problems in this thesis. The contribution in this part is a new meta-heuristic algorithm, a so-called iterative optimization, to solve a large optimization problem, the optimal results of which cannot be obtained within a reasonable time. The algorithm divides the problem into small sub-problems, which can be solved fast by standard optimization software. These sub-problems are solved sequentially and their solutions are then integrated to form the final solution to the original problem. This heuristic was later applied to solve the logical topology design problem for real-sized networks.

As one of the network planning and optimization problems, wavelength converter placement in WDM networks was first studied. We introduced an exact ILP formulation to find the nodes to place wavelength converters for a static traffic pattern. By doing experiments with a large number of different traffic patterns and using statistical methods, we could find nodes with higher probability to be equipped with wavelength converters. This gives some hints for network planners while planning WDM networks. Since it is not necessary to put wavelength converters at all nodes, one needs to find the nodes that give the highest pay-off to place wavelength converters.

In WDM networks with wavelength converters, especially limited-range converters, one has to take into account their presence while assigning wavelengths for a dynamic lightpath. This makes the wavelength assignment problem in networks with converters differ from the one in networks without converters. Therefore, we introduced a distributed algorithm to solve the wavelength assignment problem for WDM networks with sparse presence of limited-range wavelength converters. The algorithm results in an optimal wavelength path in terms of the number of converters used along the path. Consequently, it offers lower blocking probability in comparision with other approaches. The advantage of this algorithm is that it does not incur any signalling overhead while being implemented in a distributed manner. Another important finding of this work is that it is more important to save wavelength converters in the case of limited-range conversion than to save wavelength resources.

In IP over WDM networks, designing a logical topology is one of the important issues. Logical topologies must be designed to carry traffic efficiently while satisfying network resource constraints. In this thesis, we presented a new approach to design logical topologies. The problem was formulated as an MILP problem with the objective of maximizing the number of remaining lightpaths. This approach saves network resources (wavelength and transceivers) for future use because multiple different logical topologies may need to be implemented in the same WDM network. Since the MILP problem gets intractable rapidly when the network size increases, we proposed a heuristic algorithm based on the iterative optimization to solve the problem for real-sized networks.

IP traffic is subject to change over time. Hence, a logical topology, which is designed only for a specific traffic pattern, can be outdated once the traffic changes and consequently cannot carry efficiently the traffic. This requires a reconfiguration of the logical topology to adapt to the traffic variation. In fact, the reconfigurability is a valued feature of IP-over-WDM networks. In this thesis, we proposed a reconfiguration algorithm using two approaches: centralized and distributed. For the centralized approach, the problem was formulated as an ILP problem and solved by a central manager. The performance of the centralized approach was investigated and compared with the best existing one, Gencata's algorithm. The results shows that our algorithm triggers much less reconfigurations than Gencata's one. In addition, the fact that our algorithm allows multiple lightpath changes results in a better load balancing in the network. Based on the mathematical model presented in the centralized approach, we developed a distributed algorithm using the Lagrangian relaxation method. The capcacity constraint was first moved to the objective function to form a dual problem. This problem is then solved by all involved nodes in a distributed manner. The nodes used a link-state routing protocol to exchange messages with each other. This is the first time, a distributed algorithm has been developed for the logical topology reconfiguration problem. The algorithm can lend itself to a protocol implementation.

Like all types of networks, optical networks are prone to failure, but a single failure of optical networks can cause severe consequences due to the huge data loss. Therefore, survivability is a big concern while operating an optical network. In this thesis, we studied the multi-layer protection problem for IP/MPLS over WDM networks under a single link failure. Several multi-layer protection schemes have been proposed in the literature. However, the main weakness of these approaches is the high complexity with a lot of signalling overheads. In this thesis, we proposed a new protection mechanism, which is inspired by the *p-cycle* protection. In this mechanism, a protection topology, which always remains connected, is designed in advance in the IP layer. A proper number of wavelengths are reserved in this topology for protection purpose. The performance of the new protection mechanism was compared to two single layer protection approaches, i.e IP layer and WDM layer protection. The results showed that the new mechanism had a blocking probabiliy close to IP-layer protection (and much lower than WDM layer protection) and even lower if the network is highly meshed. Inherited from *p-cycle* protection, this mechanism has short restoration time since all back-up paths are switched in advance. Another important advantage of this mechanism is the fast routing for a new LSP request since only the primary path needs to be found. Its protection path has been established in advance, which significantly reduces the computation overhead.

8.2 Outlook

Eventhough this thesis delt with several different issues in optical networks using WDM technology, it didn't cover all problems. Therefore, more studies are required to make WDM networks become a real next generation network. For future works, beside revising and refining the existing solutions to the above-studied problems, one can study many extended problems in wavelength routed networks such as designing multiple logical topologies for a WDM networks, designing a survivable multi-layer network, and survivability for multi-layer WDM networks under Shared Risk Link Group (SRLG) failure or node failure.

In addition, in this thesis, we considered only unicast communication. However, more and more applications nowadays make use of multicast communication, such as multimedia, teleconferencing, distributed computation and real-time workgroups. Thus, multicast routing and wavelength assignment in WDM network will require more research.

Bibliography

[AM06] G. Agrawal and D. Medhi. Lightpath topology configuration for wavelength-routed ip/mpls networks for time-dependent traffic. In *Proceeding of IEEE GLOBECOM, San Francisco*, Nov 2006.

[Bec01] Dirk Beckmann. *Algorithmen zur Planung und Optimierung moderner Kommunikationsnetze*. PhD thesis, Hamburg University of Technology, 2001.

[BEP$^+$96] K. Bala, G. Ellinas, M. Post, Chien-Chung Shen, J. Wei, and N. Antoniades. Toward hitless reconfiguration in wdm optical networks for atm transport. In *Procedding of IEEE GLOBECOM, London*, Nov. 1996.

[BK95] A. Birman and A. Kershenbaum. Routing and wavelength assignment methods in single-hop all- optical networks with blocking. In *Proceeding of IEEE INFOCOM95, Boston*, 1995.

[BM96] D. Banerjee and B. Mukherjee. A practical approach for routing and wavelength assignment in large wavelength routed optical networks. *IEEE Journal on Selected Areas in Communications*, 14(5):903–908, 1996.

[BM00] D. Banerjee and B. Mukherjee. Wavelength routed optical networks: Linear formulation, resource budgeting tradeoffs and a reconfiguration study. *IEEE/ACM Transaction of Networking*, 8(5):598–607, 2000.

[BM05] A. Brzezinski and E. Modiano. Dynamic reconfiguration and routing algorithms for ip-over-wdm networks with stochastic traffic. *IEEE/OSA Journal of Lightwave Technology*, 23(10):3188–3205, 2005.

[BS97] R. A. Barry and S. Subramaniam. The max-sum wavelength assignment algorithm for wdm ring networks. In *Proceeding of Optical Fiber Communication Conference OFC'97*, 1997.

[CGK89] I. Chlamtac, A. Ganz, and G. Karmi. Purely optical networks for terabit communication. In *Proceeding of IEEE INFOCOM 89,Washington DC*, 1989.

[CGK92] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath communications: An approach to high-bandwidth optical wans. *IEEE Transactions on Communications*, 40(7):1171–1182, 1992.

[CL03] X. Chu and B. Li. Wavelength converter placement under differen rwa algorithms in wavelength-routed all-optical networks. *IEEE Transactions on Communications*, 51(4):607–617, 2003.

[CLRS01] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms (2nd ed.)*. MIT Press and McGraw-Hill, 2001.

[COS] COST. http://www.cost.esf.org/.

[CP99] S. Chatterjee and S. Pawlowski. All-optical networks. *Communications of the ACM*, 42(6):74–83, June 1999.

[CY94] K. Chan and T.P. Yum. Analysis of least congested path routing in wdm lightwave networks. In *Proceedings of IEEE INFOCOM, Toronto*, 1994.

[DR00] R. Dutta and G.N. Rouskas. A survey of virtual topology design algorithms for wavelength routed optical networks. *Optical Network Magazine*, 1(1):7379, 2000.

[EIS76] S. Even, A. Itai, and A. Shamir. On the complexity of timetable and multicommodity flow problems. *SIAM Journal on Computing (SIAM)*, 5(4):691–703, 1976.

[FCLT05] Y. C. Foo, S. F. Chien, A. L. Y. Low, and C. F. Teo. New strategy for optimizing wavelength converter placement. *Optics Express*, 13(2):545–551, 2005.

[FL07] C. Fang and C.P. Low. A flexible wavelength converter placement scheme for guaranteed wavelength usage. *Journal of Communications*, 2(1):34–43, 2007.

[FV00] A. Fumagalli and L. Valcarenghi. Ip restoration vs. wdm protection: Is there an optimal choice? *IEEE Network*, 14:34–41, 2000.

[GB04] W. Golab and R. Boutaba. Policy-driven automated reconfiguration for performance management in wdm optical networks. *IEEE Communication Magazin*, 42(1):44–51, 2004.

[GJ03] S. Gao and X. Jia. An optimization model for placement of wavelength converters to minimize blocking probability in wdm networks. *Journal of Lightwave Technology*, 21(3):684–694, 2003.

[GM03] A. Gencata and B. Mukherjee. Virtual adaptation for wdm mesh networks under dynamic traffic. *IEEE/ACM Transaction of Networking*, 11(2):236–247, 2003.

[Gof03] David R. Goff. *Fiber Optic Video Transmission, 1st ed.* Focal Press: Woburn, Massachusetts, 2003.

[GS98] W. Grover and D. Stamatelakis. Cycle-oriented distributed preconfiguration: ring-like speed with mesh-like capacity for self-planning network restoration. In *Proceedings of IEEE International Conference on Communications ICC'98, Atlanta*, 1998.

[GSM05] S.F. Gieselman, N.K. Singhal, and B. Mukherjee. Minimum-cost virtual topology adaptation for optical wdm mesh networks. In *Proceeding of IEEE ICC, Seoul*, May 2005.

[HL06] Q. D. Ho and M. S. Lee. Converter-aware wavelength assignment in wdm networks with limited-range conversion capability", ieice trans. commun., vol. 89(b), pp. 436-445, 2006. *IEICE Transactions on Communications*, 89(B):436–445, 2006.

[HMM97] H. Harai, M. Murata, and H. Miyahara. Performance of alternate routing methods in all-optical switching networks. In *Proceeding of IEEE INFOCOM 97, Kobe*, 1997.

[Ilo] Ilog. http://www.ilog.com/products/cplex/.

[JA96] G. Jeong and E. Ayanoglu. Comparison of wavelength-interchanging and wavelength-selective cross-connects in multiwavelength all-optical networks. In *Proceeding of IEEE INFOCOM 96, San Francisco*, 1996.

[JBA06] A. Jaekel, S. Bandyopadhyay, and Y. Aneja. Logical topology design for wdm networks using survivable routing. In *Proceedings of ICC'06, Istanbul*, 2006.

[JDH$^+$03] X. Jia, D. Du, X. Hu, H. Huang, and D. Li. On the optimal placement of wavelength converters in wdm networks. *Computer Communications*, 26:986–995, 2003.

[JS05] H.Y. Jeong and S.W. Seo. A binary linear program formulation for the placement of limited-range wavelength converters in wavelength-routed wdm networks. *IEEE/OSA Journal of Lightwave Technology*, 23(10):3076–3091, 2005.

[KA96a] M. Kovacevic and A. Acampora. Benefit of wavelength translation in all-optical clear channel networks. *IEEE Journal on Selected Areas in Communications*, 14(5):868–880, 1996.

[KA96b] M. Kovacevic and A. S. Acampora. Electronic wavelength translation in optical networks. *IEEE/OSA Journal of Lightwave Technology*, 14(6):1161–1169, 1996.

[KA98] E. Karasan and E. Ayanoglu. Effects of wavelength routing and selection algorithms on wavelength conversion gain in wdm optical networks. *IEEE/ACM Transactions on Networking*, 6(2):186–196, 1998.

[KAGS05] D.R. Karcius, W. Assis, H. Giozza, and M. Savasini. Iterative virtual topology design to maximize the traffic scaling in wdm networks. In *Procedding of the 2nd IFIP International Conference on Wireless and Optical Communication Networks (WOCN'05), Dubai*, 2005.

[KL01] M. Kodialam and T.V. Lakshman. Integrated dynamic ip and wavelength routing in ip over wdm networks. In *Proceedings of IEEE INFOCOM'01*, 2001.

[KS01] R.M. Krishnaswamy and K.N. Sivarajan. Design of logical topologies: a linear formulation for wavelength routed optical networks with no wavelength changers. *IEEE/ACM Transactions on Networking*, 9(2):186198, 2001.

[LP03] W. Ling and Y. Peida. The optimal design of logical topology with qos constraints in ip over wdm networks. In *IEEE International Conference on Communication Technology (ICCT'03), Beijing*, 2003.

[LS99] L. Li and A. K. Somani. Dynamic wavelength routing using congestion and neighborhood information. *IEEE/ACM Transactions on Networking*, 7(5):779 – 786, 1999.

[LS05] K. Lee and M.A. Shayman. Rollout algorithms for logical topology design and traffic grooming in multihop wdm networks. In *Proceedings of IEEE Globecom'05*, 2005.

[LST06] G. Lee, X. Sugang, and Y. Tanaka. A logical topology design with tabu-search in ip over wdm optical networks. In *Proceeding of Asia-Pacific Conference on Communications*, 2006.

[Mau03] Christian Mauz. p-cycle protection in wavelength routed networks. In *Proceedings of the Seventh Working Conference on Optical Network Design and Modelling (ONDM'03)*, 2003.

[MBRM96] B. Mukherjee, D. Banerjee, S. Ramamurthy, and A. Mukherjee. Some principles for designing a wide-area wdm optical network. *IEEE/ACM Transactions on Networking*, 4(5):684696, 1996.

[MCG$^+$02] S. De Maesschalck, D. Colle, A. Groebbens, C. Develder, A. Lievens, P. Lagasse, M. Pickavet, P. Demeester, F. Saluta, and M. Quagliatti. Intelligent optical networking for multilayer survivability. *IEEE Communications Magazine*, 40(1):42–49, 2002.

[MG02] C. Siva Ram Murthy and Mohan Gurusamy. *WDM Optical Networks: Concept, Design and Algorithms.* Prentice Hall PTR, 2002.

[Muk97] B. Mukherjee. *Optical communication Networks.* McGraw-Hill, New York, 1997.

[Muk00] B. Mukherjee. Wdm optical communication network: progress and challenges. *IEEE Journal on Selected Areas in Communications*, 18:18101824, 2000.

[Mul05] Eueung Mulyana. *Efficient Planning and offline routing approaches for IP networks.* PhD thesis, Hamburg University of Technology, 2005.

[NTM00] A. Narula-Tam and E. Modiano. Dynamic load balancing in wdm packet networks with and without wavelength constraints. *IEEE Journal of Selected Areas in Communication*, 18(10):1972–1979, 2000.

[PM04] M. Pioro and D. Medhi. *Routing, Flow and Capacity Design in Communication and Computer Networks*. Morgan Kaufmann (Elsvier), 2004.

[RA95] G. N. Rouskas and M. H. Ammar. Dynamic reconfiguration in multihop wdm networks. *Journal of High Speed Networks*, 4:221–238, 1995.

[Ram98] S. Ramamurthy. *Optical Design of WDM Network Architectures*. PhD thesis, University of California, Davis, 1998.

[RM98] S. Ramamurthy and B. Mukherjee. Fixed-alternate routing and wavelength conversion in wavelength-routed optical networks. In *Proceeding of IEEE GLOBECOM98, Sydney*, 1998.

[RMZ05] K. Ratnam, G. Mohan, and Luying Zhou. Differentiated qos routing of restorable sub-lambda connections in ip-over-wdm networks using a multi-layer protection approach. In *Proceedings of IEEE/CreateNet Broadnets, Boston*, 2005.

[RS95] R.Ramaswami and K.N. Sivarajan. Optimal routing and wavelength assignment in all-optical networks. *IEEE/ACM Transactions on Networking*, 3(5):489–500, 1995.

[RS96] R. Ramaswami and K.N. Sivarajan. Design of logical topologies for wavelength-routed optical networks. *IEEE Journal on Selected Areas in Communications*, 40(1):840851, 1996.

[RS03] S. Ramamurthy and L. Sahasrabuddhe. Survivable wdm mesh networks. *IEEE/OSA Journal of Lightwave Technology*, 21(4):870–883, 2003.

[RVC01] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol label switching architecture. In *RFC 3031*. IETF, 2001.

[RZG06] K. Ratnam, L. Zhou, and M. Gurusamy. Efficient multi-layer operational strategies for survivable ip-over-wdm networks. *IEEE Journal on Selected Areas in Communications*, 24(8):16–31, 2006.

[SB97] S. Subramaniam and R. A. Barry. Wavelength assignment in fixed routing wdm networks. In *Proceeding of IEE ICC 97, Montreal*, 1997.

[Sch05] D. Schupke. Automatic protection switching for p-cycles in wdm networks. *Optical Switching and Networking (OSN), Elsevier*, 2(1):35–48, 2005.

[Sch06] D. Schupke. Analysis of p-cycle capacity in wdm networks. *Photonic Network Communications, Springer*, 12(1):41–51, 2006.

[SK04] N. Sengezer and E. Karasan. A tabu search algorithm for sparse placement of wavelength converters in optical networks. *Lecture Notes in Computer Science*, 3280:247–256, 2004.

[SL05] M. Saad and Z. Luo. Reconfiguration with no service disruption in multifiber wdm networks. *IEEE/OSA Journal of Lightwave Technology*, 23(10):3092–3104, 2005.

[SM05] S. Sinha and C. S. Ram Murthy. Information theoretic approach to traffic adaptive wdm networks. *IEEE/ACM Transaction of Networking*, 13(4):881–894, 2005.

[SMS96] S. Subramaniam, M.Azizoglu, and A. K. Somani. All optical networks with sparse wavelength conversion. *IEEE/ACM Transactions on Networking*, 4(4):544–557, 1996.

[SRM02] L. Sahasrabuddhe, S. Ramamurthy, , and B. Mukherjee. Fault management in ip-over-wdm networks: Wdm protection versus ip restoration. *IEEE Journal on Selected Areas in Communications*, 20(1):21–33, 2002.

[SSG03] Schupke, Scheffel, and Grover. Configuration of p-cycles in wdm networks with partial wavelength conversion. *Photonic Network Communications, Kluwer Academic Publishers*, 6(3):239–252, 2003.

[TK07a] P.N. Tran and U. Killat. Design of logical topology for ip over wdm networks: network performance vs. resource utilization. In *Proceedings of 3rd International Network Optimisation Conference (INOC'07), Spa*, 2007.

[TK07b] P.N. Tran and U. Killat. Efficient logical topology design for ip overwdm backbone networks: Milp and heuristic approach. In *Proceedings of International*

Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'07), San Diego, 2007.

[TK08a] P.N. Tran and U. Killat. An exact ilp formulation for optimal wavelength converter usage and placement in wdm networks. In *Proceedings of GLOBECOM '08, New Orleans*, 2008.

[TK08b] P.N. Tran and U. Killat. Resource efficient logical topology design for ip-over-wdm backbone networks. *Computer Communications*, 31(16):3771–3777, 2008.

[TK08c] P.N. Tran and U. Killat. Dynamic reconfiguration of logical topology for wdm networks under traffic changes. In *Proceeding of the 11th IEEE/IFIP Network Operation and Management Symposium (NOMS), Salvador de Bahia*, April 2008.

[TK09] P.N. Tran and U. Killat. Distributed algorithm for dynamic logical topology reconfiguration in ip over wdm networks. In *Proceedings of the 14th IEEE International conference on Computer Communications ISCC'09, Sousse*, 2009.

[UQBL06] S. Uhlig, B. Quoitin, S. Balon, and J. Lepropre. Providing public intradomain traffic matrices to the research community. *ACM SIGCOMM Computer Communication Review*, 36(1):8386, 2006.

[VSK99] K. R. Venugopal, M. Shivakumar, and P. S. Kumar. A heuristic for placement of limited range wavelength converters in all-optical networks. In *Proceeding of INFOCOM'99*, 1999.

[Xin07] Chunsheng Xin. Dynamic traffic grooming in optical networks with wavelength conversion. *IEEE Trans. On Selected Area in Communications, Supplement on Optical Communication and Networking*, 25(9):50–57, 2007.

[XL99] G. Xiao and Y.W. Leung. Algorithms for allocating wavelength converters in all-opyical networks. *IEEE/ACM Trans. Networking*, 7:545–557, 1999.

[YADA01] Yinghua Ye, Chadi Assi, Sudhir Dixit, and M. A. Ali. A simple dynamic integrated provisioning/protection scheme in ip over wdm networks. *IEEE Communication Magazine*, 39(11):174–182, 2001.

[ZD02] H. W. Zhang and A. Durresi. Differentiated multi-layer survivability in ip/wdm networks. In *Proceeding of the 8th IEEE/IFIP NOMS02, Florence*, April 2002.

[ZH03] D. Zhemin and M. Hamdi. On the management of wavelength converter allocation in wdm all-optical networks. In *Proceeding of IEEE GLOBECOM, San Francisco*, Dec 2003.

[ZJM00] H. Zhang, J. P. Jue, and B. Mukherjee. A review of routing and wavelength assignment approaches for wavelength-routed optical wdm networks. *Optical Networking Magazine*, 1(1):47–60, 2000.

[ZM03a] Q. Zheng and G. Mohan. An efficient dynamic protection scheme in integrated ip/wdm networks. In *Proceedings of ICC '03*, 2003.

[ZM03b] Q. Zheng and G. Mohan. Protection approaches for dynamic traffic in ip/mpls-over-wdm networks. *IEEE Communications Magazine*, 41(5):24–29, 2003.

[ZMT05] Y. Zhang, M. Murata, and H. Takagi. Traffic-based reconfiguration for logical topologies in large-sclae wdm optical networks. *IEEE/OSA Journal of Lightwave Technology*, 23(10):2854–2867, 2005.

[ZQ98] X. Zhang and C. Qiao. Wavelength assignment for dynamic traffic in multi-fiber wdm networks. In *Proceeding of the 7th International Conference on Computer Communications and Networks, Lafayette, LA*, 1998.

[ZVZO03] A. Zalesky, H.L. Vu, M. Zukerman, and I. Ouveysi. A framework for solving logical topology design problems within constrained computation time. *IEEE Communication Letters*, 7(10):499 – 501, 2003.

[ZY02] H. Zhang and O. Yang. Finding protection cycles in dwdm networks. In *Proceedings of IEEE Int. Conf. on Communications, New York*, 2002.

[ZY07] Z. Zhang and Y. Yang. On-line optimal wavelength assignment in wdm networks with shared wavelength converter pool. *IEEE/ACM Transactions on Networking*, 15(1):234–245, 2007.

[ZZB05] Z. Zhang, W.D. Zhong, and S.K. Bose. Dynamically survivable wdm network design with p-cycle based pwce. *IEEE Communication Letters*, 9:756–758, 2005.

[ZZM04] Z. Zhang, W-D. Zhong, and B. Mukherjee. A heuristic method for design of survivable wdm networks with p-cycles. *IEEE Communication Letters*, 8:467–469, 2004.

Curriculum Vitae

Name	Tran Phuong Nga
Date of birth	19.11.1979
Place of birth	Hanoi - Vietnam

09/1994 - 06/1997	Kim Lien High School - Hanoi - Vietnam
09/1997 - 05/2002	Hanoi University of Technology, Hanoi, Vietnam Bachelor of Science in Electronics and Telecommunications
10/2003 - 02/2006	Hamburg University of Technology, Hamburg, Germany Degree: Master of Science in Information and Communication Systems Northern Institute of Technology, Hamburg, Germany Degree: MBA in Technology Management
08/2004 - 01/2005	Internship at Siemens VDO, Regensburg
03/2006 - 05/2010	Research assistant and PhD candidate Hamburg University of Technology, Hamburg, Germany Institute for Communication Networks
09/2010 - now	Postdoctoral Researcher Hamburg University of Technology, Hamburg, Germany Institute for Communication Networks