Traffic Shaping for Contention Control in OBS Networks

vorgelegt von
Master of Science
Ahmad Rostami

von der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
– Dr.-Ing. –

genehmigte Dissertation

Promotionsausschuss:
Vorsitzender: Prof. Dr. Axel Küpper
Berichter: Prof. Dr.-Ing. Adam Wolisz
Berichter: Prof. Dr. Maurice Gagnaire

Tage der wissenschaftlichen Aussprache: 11.11.2010

Berlin 2010

D83

Abstract

Traffic volume in the Internet continues to grow exponentially, which necessitates a transport network that can adapt to the increasing bandwidth requirements accordingly. The technology of choice for realizing such a transport network is optics. Nonetheless, optics does not provide a good support for packet switching, which has proven to be the most efficient method for transporting the varying traffic of the Internet. Specifically, there are three major issues hindering realization of an all-optical packet switching approach in the near future: low speed of all-optical switches, immature all-optical processors and lack of true optical buffers. One promising approach to addressing these issues is optical burst switching (OBS). OBS consists in grouping packets into bursts and out-of-band signaling that collectively eliminate the need for fast optical switches and all-optical processors. It fails however to fully eliminate the problem of buffering in the network. In fact, buffers are needed in packet switched networks for two purposes. First, they are used to keep the packets in nodes while the node controllers read and process the packets' headers. Additionally, buffers are the main tools to mitigate the contention problem over the output ports of packet switches. Although OBS eliminates the need to buffers with regards to their first role, it does not provide an efficient solution for contention resolution.

In this dissertation we aim at addressing the burst contention problem in OBS networks taking a traffic shaping approach. The work is presented in three parts. In the first part, a review of major IP over WDM techniques including OBS is presented and the contention problem associated to OBS networks is detailed. The second part presents a thorough analysis and parameterization of traffic generated at the ingress edge of the OBS network. This includes both analytical modeling of the traffic as well as simulation experiments based on real packet traces. In the analytical part, making some simplifying assumptions about the packet arrivals to the assembly unit, we model the traffic at the output of a single burst assembly queue based on the burst assembly parameters. The developed models are then utilized to study the impact of burst assembly on contention resolution inside the network. It is concluded that in order to address the burst contentions through traffic shaping the focus should be put into aggregating traffic at the edge of the network into flows with higher capacities and thereby reducing the number of traffic flows inside the networks. At the end of this part we carry out sets of simulation experiments based on real measurement-based IP packet traces to validate the results of our analytical analysis.

Based on the insights gained in the second part of the dissertation, in part three we develop an advanced traffic shaping mechanism and incorporate it into the OBS network architecture. This results in the introduction of a novel architecture for optical transport network that we call it virtual optical bus (VOB). VOB is an evolutionary

architecture based on OBS, in which several traffic flows are grouped into clusters and within each cluster a form of coordination (shaping) on burst transmission is introduced. Specifically, different traffic sources belonging to the same cluster inject their bursts into the network in such a way that burst collision within the cluster is avoided. Additionally, clustering of flows and selection of paths for clusters are done in a way that the interaction among routes of clusters in the network is minimized. This leads to a large reduction of burst collisions in the network and also an increase in the network throughput. This comes at the cost of marginal increase in the delay that bursts experience at ingress edge of the network.

Zusammenfassung

Der Internet-Datenverkehr wächst exponentiell, was ein Transportnetz erfordert, das sich an die steigenden Bandbreitenanforderung anpassen kann. Optische Datenübertragung ist die bevorzugte Technologie für die Realisierung solch eines Transportnetzes. Allerdings bietet optische Datenübertragung keine gute Unterstützung für die Paketvermittlung, die sich als das effizienteste Verfahren für den Transport des variablen Datenverkehr im Internet erwiesen hat. Es gibt drei konkrete Faktoren, die die Realisierung einer rein optischen Paketvermittlung in der nahen Zukunft behindern: Geringe Geschwindigkeit rein optischer Schalter, rudimentäre rein-optische Prozessoren und der Mangel an echten optischen Puffern. Ein vielversprechender Ansatz zur Bewältigung dieser Probleme ist die optische "Burst"-vermittlung (Optical Burst Switching (OBS)). OBS fasst mehrere Pakete in Bündeln (Bursts) zusammen und nutzt zusätzlich eine Out-of-Band-Signalisierung, so dass schnelle optische Schalter und rein-optische Prozessoren nicht mehr notwendig sind. Dieser Ansatz vermag es jedoch nicht, das Problem der Pufferung im Netz vollständig zu lösen. Puffer werden in paketvermittelten Netzen für zwei Zwecke benötigt: Erstens, um die Pakete im Knoten zu halten, während der Controller den Header des Paketes liest und verarbeitet. Darüber hinaus sind Puffer die wichtigsten Instrumente um Kollisionen der Pakete an den Ausgängen der Knoten aufzulösen. OBS stellt zwar eine Lösung für das erste Problem dar, das Verfahren bietet jedoch keine effiziente Lösung für das zweite Problem.

Diese Dissertation befasst sich mit der Kollisionsauflösung in OBS Netzwerken auf der Basis eines Traffic-Shaping-Ansatzes. Die Arbeit besteht aus drei Teilen. Im ersten Teil wird ein Überblick über die wichtigsten IP-über-WDM-Techniken einschließlich OBS präsentiert und die Kollisionsauflösung in Hinblick auf OBS Netze beschrieben. Der zweite Teil enthält eine gründliche Analyse und Parametrierung des Verkehrs wie er am Eintrittspunkt eines OBS-Netzwerkes generiert wird. Dies umfasst sowohl eine analytische Modellierung des Verkehrs als auch Simulationen, die unter Zuhilfenahme von Ergebnissen realer Messungen durchgeführt wurden. Im analytischen Teil wird mithilfe von vereinfachenden Annahmen über Paketankunftszeiten der Verkehr am Ausgang einer einzelnen Burst-Aggregation-Warteschlange (Burst Assembly Queue) in Abhängigkeit von Burst-Aggregation Parametern modelliert. Die entwickelten Modelle werden dann verwendet, um die Auswirkungen von Burst-Aggregation auf Kollisionsauflösung zu studieren. Eine Schlussfolgerung ist, dass sobald Traffic-Shaping Strategien eingesetzt werden, das Hauptaugenmerk auf der Zusammenfassung des Datenverkehrs am Rande des Netzwerkes in Datenströme mit höheren Kapazitäten liegen sollte, um die Anzahl der Datenströme im Netzwerken zu reduzieren. Am Ende dieses Teils führen wir eine Reihe von Simulationen auf Grundlage von IP-basierten Messergebnissen durch, um die Resultate der analytischen Verfahren zu validieren.

Aufbauend auf den Erkenntnissen aus dem zweiten Teil der Dissertation, entwickeln wir im dritten Teil einen erweiterten Traffic-Shaping-Mechanismus und integrieren ihn in die OBS Netzwerkarchitektur. In diesem Zusammenhang führen wir eine neuartigen Architektur für optische Netzwerke ein, den sogenannten virtuellen optischen Bus (Virtual Optical Bus, VOB). VOB ist eine evolutionäre OBS-Architektur, in der mehrere Verkehrsströme in Gruppen (Cluster) eingeteilt werden und innerhalb der einzelnen Gruppen eine Form der Koordinierung (Shaping) der Burst-Übertragung eingesetzt wird. Gehören verschiedene Quellen zu dem gleichen Cluster, wird deren Last in das Netzwerk so eingeführt, dass Kollisionen innerhalb des Clusters vermieden werden. Darüber hinaus werden einzelne Datenströme so zusammengeführt bzw. die Wegewahl pro Cluster so durchgeführt, dass die Interaktion zwischen den Wegen der einzelnen Cluster im Netzwerk minimiert wird. Dies führt sowohl zu einer deutlichen Reduzierung der Burst-Kollisionen im Netz als auch zu einer Erhöhung des Netzwerk-Durchsatzes. Es geschieht auf Kosten einer marginalen Zunahme der Verzögerung der Bursts am Rand des Netzwerks.

Acknowledgments

I would like to thank everyone who helped me during my doctoral study. First, I am very grateful to my advisor Prof. Adam Wolisz for his support, guidance and valuable criticism that he provided throughout the course of this work. I am also thankful to all my friends, and my colleagues at Telecommunication Networks Group (TKN), and I am forever indebted to my parents for their continual understanding and encouragement throughout all my years of study.

Contents

List of Figures

List of Tables

List of Abbreviations

ADM	Add Drop Multiplexer
AS	Autonomous System
ATM	Asynchronous Transfer Mode
BHP	Burst Header Packet
CapEx	Capital Expenditures
CCDF	Complementary Cumulative Distribution Function
CDF	Cumulative Distribution Function
CoV	Coefficient of Variation
DWRON	Dynamic Wavelength Routed Optical Network
DXC	Digital Cross Connect
FDL	Fiber Delay Line
FGN	Fractional Gaussian Noise
FIFO	First In First Out
FR	Frame Relay
ICT	Information and Communication Technology
ILP	Integer Linear Programming
IP	Internet Protocol
JET	Just Enough Time
JIT	Just In Time
KSP	K-Shortest Path

LAUC	Latest Available Unscheduled Channel
LAUC-VF	Latest Available Unscheduled Channel with Void Filling
LRD	Long Range Dependency
LSP	Label Switched Path
MAC	Medium Access Control
MILP	Mixed Integer Linear Programming
MPLS	Multi Protocol Label Switching
MRA	Multiresolution Analysis
O-D	Origin Destination
O/E/O	Optical Electrical Optical Conversion
OADM	Optical Add Drop Multiplexer
OAM	Operation, Administration and Management
OAMP	Optical Amplifier
OBS	Optical Burst Switching
OpEx	Operational Expenditures
OPS	Optical Packet Switching
OXC	Optical Cross Connect
PCM	Pulse Code Modulation
PDF	Probability Density Function
QoS	Quality of Service
R&D	Research and Development
RAM	Random Access Memory
ROADM	Reconfigurable Optical Add Drop Multiplexer
RWA	Routing and Wavelength Assignment
SDH	Synchronous Digital Hierarchy

SOA	Semiconductor Optical Amplifier
SONET	Synchronous Optical Networking
TCP	Transmission Control Protocol
TDM	Time Division Multiplexing
TWC	Tunable Wavelength Converter
VC	Virtual Circuit
VDQ	Virtual Destination Queue
VOB	Virtual Optical Bus
VoD	Video on Demand
VoIP	Voice over IP
WDM	Wavelength Division Multiplexing
WRON	Wavelength Routed Optical Network

Chapter 1

Introduction

A large amount of efforts and investments has been put into investigating different networking architectures and protocols until the communication network community came to the conclusion that the Internet protocol (IP) [2] is the layer where data traffic of all forms and in the not-too-distant future also digital voice and multimedia traffic will converge. Increasing number of TCP/IP [3] networking applications developed every year as well as the large number of new techniques proposed with the objective of enabling transport of voice and multimedia traffic over IP are all evidences of this trend. In other words, the communication networks are becoming more and more IP centric. On the other hand, the data traffic has been exponentially increasing over the last decade. The most recent statistical estimates show around 40-50% annual increase in the internet traffic [4], which is a considerable growth rate. These two facts lead to the key question "what technology and architecture have the potential to transport the ever-increasing traffic generated at the IP layer?"

This is the point where transport networks for IP come into play [5]. A transport network here refers to a telecommunications infrastructure that serves as a substrate through which the data generated by the IP network is transferred between the IP network nodes. Any proposal for the transport network for IP traffic must have several features. Foremost among these is the ability to provide a large capacity to support different bandwidth requirements of the IP network. Beyond that, cost of implementing or upgrading the transport network, i.e., capital expenditures (CapEx), and also that of operation and maintenance of the network, i.e., operational expenditures (OpEx), have to be as low as possible. Other requirements include efficient use of the resources (network utilization), reliability as well as the support for quality of service (QoS). To make efficient use of the networking resources, it might be useful that the transport network further supports the dynamic assignment of capacity to different services, i.e. to cope with the time-varying nature of IP traffic. All these requirements coupled with the technological developments in the field of communications give rise to the rapid changes in transport networks as we are witnessing today.

On the other hand and in parallel to the rapid developments in the communication

networking arena, the optical technology has evolved and achieved a maturity that makes it the technology of choice for building new generation of transport networks. Today, using the wavelength division multiplexing (WDM) technique, one can transmit hundreds of gigabits per second on a single strand of fiber optic over distances as large as hundreds of kilometers. In addition, both cost and power consumption of an optical communication system are smaller than those of a system based on electronics [6]. All these unique features have left almost no doubt that the WDM technology is the key enabling technology for design and implementation of high-capacity transport network for IP traffic – i.e., through IP over WDM. Over the past 15-20 years several proposals have been presented for building transport networks based on the optical technology [7].

Fig. 1.1 depicts the evolution of WDM networks over the time, where traffic granularity refers to both the volume of the traffic and size of data units [1]. In the first generation of optical networks, WDM is used merely as a physical layer transmission method; that is, switching and routing of traffic in the network are left to the sophisticated electronic layer. For switching and routing of data, optical networks of the first generation rely on digital cross connects (DXC). The complexity and costs of DXCs increases with the transmission rate of the WDM links and as a result DXCs turn into a bottleneck of the network over the time. On the other hand, the technological progresses over time have enabled the WDM layer to subsume some of the networking functionalities of higher layers such as routing and multiplexing by introducing optical amplifiers (OAMP) and optical add/drop multiplexers (OADM) [6]. These progresses helped the emergence of second generation optical networks, which are based on optical cross-connects (OXC) [6]. An OXC realizes switching and routing of WDM channels among the WDM links that are connected to it.

From the networking perspective, both first and second generation optical networks are centered on the connection-oriented operation. As such, they both provide transport service to the higher layer in the form of optical circuits of fixed capacities. In the first generation the fixed capacity is usually a dedicated time-slot, which allows for multiplexing and demultiplexing of data streams of different bitrates through DXCs in the network. This approach suffers from several disadvantages, namely, lack of scalability of DXCs at very high data rates, lack of transparency of the optical circuit with respect to datarates and modulation format and the very long times needed for establishment/tearing down of a circuit. In the second generation networks optical circuits are established in form of lightpaths, which is a circuit that spans over several links and is routed by several OXCs [8]. An optical transport network based on lightpaths is called wavelength routed optical networks (WRON) [9, 10]. In this approach the whole capacity of a wavelength channel – being either 2.5 (OC-48), 10 (OC-192) or 40 (OC-678) Gb/s – is dedicated to a lightpath. Migrating from first generation to second generation optical networks brings several advantages. Specifically, the lightpath is transparent both in terms of datarates and modulation technique. Also, the lightpaths can be set up much faster than the optical circuits in the first generation networks.

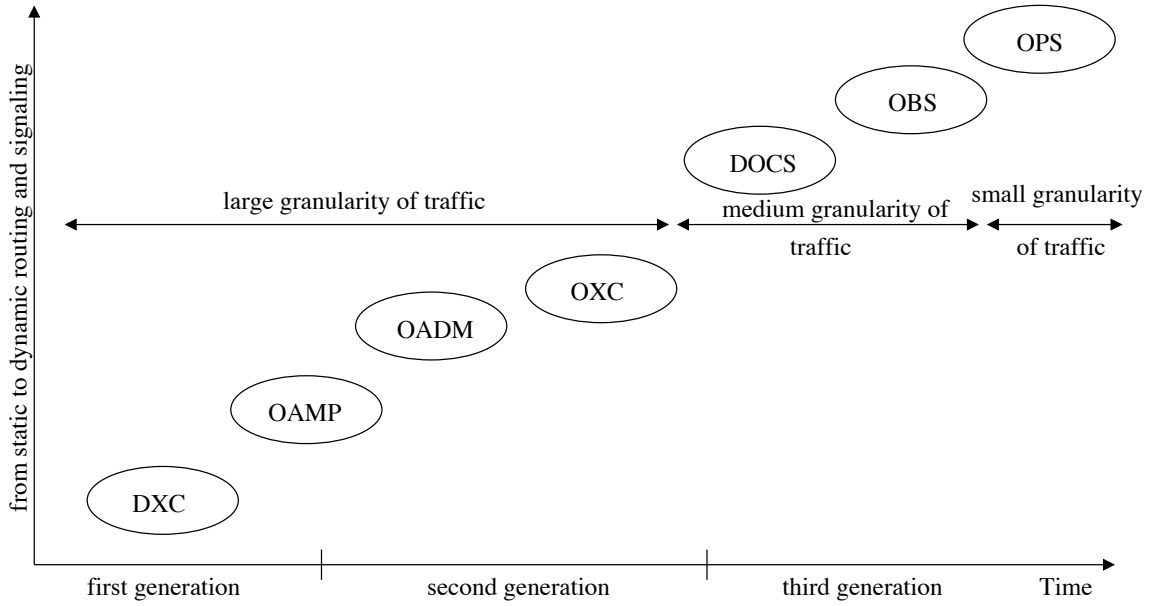

Figure 1.1: Evolution of optical networks over the time [1].

These features together result in a lower CapEx and OpEx for second generation optical transport networks as compared to the first generation. Nevertheless, the WRON networks suffer from under-utilization of network capacity, which causes wasting of the resources. In addition, they are not fast enough to cope with the dynamic of the IP traffic.

Recently design and prototyping high-speed OXCs in R&D laboratories have motivated the optical network research community to propose a third generation of optical transport networks aiming at addressing the shortcomings of the second generation networks [11]. The main focus in the third generation is to increase the flexibility of networks in allocating resources in order to cope with the variation of traffic generated by the higher layer protocol, i.e., IP. Third generation networks encompass a variety of architectures and proposals ranging from dynamic wavelength routed optical networks (DWRON) [12, 13, 14] to optical packet switching (OPS) [15, 16, 17, 18, 19].

In DWRON networks lightpaths are established and removed in a dynamic manner and based on varying requirements of the IP layer. The main constraint in this approach is the duration of an established circuit that should be much larger than the time it takes to set up the lightpath, otherwise, the resource utilization in the network would be low. On the other side of the spectrum of third generation optical networks there is OPS, which is a quite ambitious and futuristic approach. OPS operation is based on optically transmitting packets through the transport network, where switching and routing of individual packets along their paths to the egress edge node is performed all-optically. OPS is a natural approach taking into account that under highly dynamic

traffic like IP, packet switching has proven to be the most efficient switching paradigm in the sense that it has a smaller networking resources requirement when compared to circuit switching, though it comes at the cost of higher switching and processing requirements. Nevertheless, packet switching has some requirements that hinder its realization in the optical domain in the short term. More specifically, implementation of a packet switched transport network necessitates large processing and buffering capacities as well as ultra fast switching matrices all in the optical domain. These requirements, however, cannot be met easily by the optical technology in the near future and require breakthroughs in this field.

Therefore, there is a need to devise an optical transport network that on one hand overcomes the inefficiencies of the WRON in terms of the flexibility and resource utilization, and on the other hand does not have the stringent requirements of the OPS. Optical burst switching (OBS) is an architecture that has been proposed to address these issues [20, 21, 22, 23, 24].

In fact, OBS is an intermediate switching paradigm – between WRON and OPS – that enables dynamic allocation of resources in the optical domain at sub-wavelength granularity, thereby achieving a high utilization comparable to that of OPS while avoiding its requirements. In OBS the switching is performed at the level of burst – as compared to packet in OPS – which is a jumbo packet created at the edge of the network from merging several data packets having the same destination in the transport network. This process is called burst assembly and is one of the major principles of OBS. The use of jumbo packets instead of small size packets enable the realization of a packet switched network without requiring very high-speed OXCs. This further relaxes the processing required at the control unit of each switch, because only one header needs to be processed for each jumbo packet comprising several IP packets. Accordingly, the processing of packet headers in intermediate nodes can be handled by electronics. To facilitate that, signaling and data communications in OBS are carried out on separate channels, i.e., out-of-band signaling scheme is used. This eliminates the need for optical-electrical-optical (O/E/O) conversion at any intermediate node in the network in the data path. Specifically, a separate control packet carries the header information of each data burst. This control packet is transmitted on one of the signaling channels, which are the only channels that go through O/E/O conversion in intermediate nodes.

Although OBS overcomes the main hindering issues associated with OPS, there is still an issue that needs to be addressed before it can be viable for commercial deployment. That is namely the packet contention problem – also known as collision problem[1] – that can frequently occur on output links of OBS nodes in the network. A collision occurs when two or more data packets should use the same channel on a given link for an overlapping period of time. In electronic packet-switched networks, efficient avoidance of this phenomenon is usually done by introducing buffers in switches. These

[1]Throughout this dissertation contention and collision are used interchangeably.

buffers hold contending packets during times of contentions. This approach, however, cannot be readily adopted in OBS since there is no equivalent to random access memories in the optical domain. Consequently, the burst contentions problem has become central to the design and proper operation of OBS networks.

1.1 Ideas and Approach

This thesis focuses on the burst collision problem inside the OBS network, which has been at the center of research activities related to OBS networks over the last ten years. A variety of solutions has been proposed for this purpose, ranging from wavelength conversion to FDL buffering to deflection routing, as detailed in Chapter 3. Most of the proposed solutions follow a reactive approach to addressing the burst collisions, that is, they focus on resolving the collisions only after they occur. Instead, in this thesis we take a proactive approach aiming at avoiding/minimizing the occurrence of burst contentions inside the OBS network. The thesis is based on our hypothesis that the collisions can be effectively avoided/reduced by means of traffic shaping. At the edge of the OBS network, where data packets are still in the electrical domain, inexpensive electrical buffers can be utilized to shape the traffic that is injected into the network in such a way that the burst collisions inside the network are controlled. In fact, the OBS network does perform a special form of traffic shaping at the edge through implementing the burst assembly process. As such, the burst assembly should also be considered as a built-in traffic shaping suite for the OBS network.

Therefore, it is of great importance that as our first step look at the shaping effects of the burst assembly process. For this purpose we adopt one of the most general burst assembly algorithms that has two control parameters: size and time. While the the former one controls the size of generating bursts, the latter one guarantees that the assembly delay is not excessively large. These two parameters together with the statistical characteristics of the traffic that feeds the burst assembler collectively characterize the traffic that is injected into the network by an OBS ingress node. That is, for a given input traffic varying the assembly parameters can influence the statistical characteristics of the generated bursts such as burst size and burst interdeparture time distributions. This raises two interesting questions as follows. First, what does the traffic at the output of the burst assembly look like and what is the range of the burst traffic characteristics, e.g. burst length variation, that can be achieved through varying the burst assembly control parameters? Second, what are the impacts of changing the burst traffic characteristics – through the assembly process – on the burst contentions problem inside the OBS network?

To address these questions we start with an analytical approach and make some simplifying assumptions concerning the characteristics of traffic at the input of the burst assembler. This allows us to exactly parameterize – based on the assembly control parameters – the distributions of size and interdeparture time of bursts that

are generated by the burst assembler. The parametric models developed in this way are utilized to study the characteristics of burst level traffic and to show that the variability of burst sizes and burst interdeparture times can be shaped through varying the assembly parameters, which in turn influence the burst contentions among burst flows generated by different burst assemblers inside the OBS network. Specifically, based on the developed models we set up and perform sets of experiments to investigate the impacts of the traffic shaping introduced by the burst assembly on the burst collision rates inside the OBS network. The results lead, among others, to the conclusion that in order to reduce burst contentions inside the OBS network through traffic shaping the efforts should be focused on decreasing the number of traffic flows inside the network through aggregating traffic into flows with higher capacities at the edge of the network.

In order to show the validity of the analysis carried out based on the hypothetical traffic models as input of the burst assembler, we next turn to the simulation-based parameterization of the burst traffic characteristics using real measurement-based IP packet traces. This brings two benefits. First, the results throw light on statistical characteristics of the traffic at the output of the burst assembly unit under realistic traffic. Second, it demonstrates that the traffic shaping effects of the burst assembly and its impact on the burst contentions under realistic input traffic are in agreement with those achieved under simplistic traffic model.

The parameterization of the traffic at the output of the burst assembly unit provides with very useful guidelines regarding the opportunities and limitations of the built-in traffic shaping suite of the OBS network. According to one of these guidelines, the burst collision rate can be largely – in the range of several orders of magnitude – reduced by aggregating traffic into flows with higher capacity and thereby reducing the total number of flows inside the network. We exploit this guideline as a basis for developing additional traffic shaping functionalities with the objective of burst collision avoidance/minimization inside the OBS network.

Through developing an advanced traffic shaping mechanism and incorporating it into the OBS network architecture, we put forward a novel architecture for optical transport network and term it virtual optical bus (VOB). VOB is an evolutionary architecture based on OBS, in which several traffic flows – each generated by a single burst assembler – are grouped into clusters and within each cluster a form of coordination – shaping – on burst transmission is introduced. More specifically, different traffic sources belonging to the same cluster inject their bursts to the network in such a way they do not collide with each other in the network, i.e. this coordination assures a collision-free burst transmission within each cluster. To allow for this kind of burst-level coordination, a new medium access control (MAC) protocol is developed and incorporated to each source node in the network. Additionally, clustering of flows and selection of paths for clusters are done in a way that the interaction among routes of clusters in the network is minimized. This leads to a large reduction of burst collisions in the network and also an increase in network throughput. This comes at a cost of marginal increase in the delay that bursts experience at ingress edge of the network.

1.2 Structure of Thesis

This thesis is presented in three parts. The first part provides an overview of the transport networks including OBS and the architecture of the system under consideration:

Chapter 2 – Transport Networks for IP Traffic An overview of the main approaches to transport of IP traffic based on WDM technology is presented in this chapter. In addition, the advantages and disadvantages of each solution are discussed and a qualitative comparison among these proposals is presented.

Chapter 3 – Optical Burst Switching In this chapter, the optical burst switching paradigm is introduced and the most relevant architectural aspects and protocols are detailed. Among others, contention control techniques in OBS networks are introduced and discussed in detail. At the end of this chapter, we formulate the goal of the thesis and introduce the system architecture as well as the performance metrics that are considered.

The second part of the dissertation presents the parameterization of the traffic at the output of the burst assembly unit and investigates traffic shaping effects of the burst assembly:

Chapter 4 – Burst Assembly Process: An Analytical Investigation In this chapter we take an analytical approach to investigate the shaping effects of burst assembly unit on the traffic. In addition, we study the impact of the assembly parameters on the collision problem inside the network.

Chapter 5 – Burst Assembly Process: A Trace-Based Investigation This chapter presents a simulation-based parameterization of the burst traffic characteristics based on the real IP packet traces collected from a core link in the Internet. In addition, by incorporating real traffic traces in a set of simulation experiments, the validity of observations in Chapter 4 regarding the positive influence of the burst assembly process on burst collisions inside the network are examined.

Finally, part three presents our traffic-shaping-based method, which is designed to address the burst collision problem inside the OBS network:

Chapter 6 – Virtual Optical Bus A novel evolutionary optical transport network based on OBS is presented. In addition to formal description of elements of VOB architecture, MAC protocol design and formulation of the VOB network design are presented in this chapter.

Chapter 7 – Performance Analysis of VOB Performance of the VOB architecture presented in Chapter 6 is investigated using discrete event simulation. The results presented in this chapter demonstrate the merits of the VOB framework in avoiding packet collisions in the network as well as in improving overall network throughput.

Chapter 8 – VOB Network Design Examples This chapter presents two design examples of the VOB network. In addition, performance of designed networks are investigated and compared to results of the same networks operated under classical OBS.

Chapter 9 – Conclusion The achievements and contributions made throughout this work are summarized in this chapter. Additionally, open issues regarding the VOB architecture are outlined.

Part I

Background Material

Chapter 2

Transport Networks for IP Traffic

In the last 10 years we have been witnessing a persistent increase in the amount of data exchanged over the Internet, i.e. the bandwidth demand. Following recent studies by the Minnesota Internet traffic studies group in US (MINTS) [4], the monthly amount of traffic exchanged over the Internet in the world at the year-end 2009 is estimated to be around 7,500-12,000 PB[1]. In fact, Internet has experienced different rates of increase in amount of traffic during different time periods, as a result of the introduction of new networking applications and the shifts in the usage patterns of Internet users. For example, during 1995-1996, Internet traffic experienced a growth rate of 100% every 100 days in the US due to the world wide web boom, see [4, 25]. The latest estimates show around 40-50% annual growth in Internet traffic at the year-end 2009 and more or less the same situation is projected for the years to come [4]. This trend has been fueled mainly by the introduction and increasing popularity of new bandwidth-demanding applications to the Internet such as voice over IP (VoIP) [26], video broadcast based on IP (e.g., IPTV [27]), video on demand (VoD) [28], network gaming and peer-to-peer networking applications [29] (e.g., multimedia file exchange). Although most of these applications are rather new to the Internet, they take a large share of the whole Internet capacity. For instance, according to some estimates, almost 70% of the traffic flowing in the Internet belongs to the peer-to-peer applications [30]. This has further led to a large variability of traffic patterns over the Internet.

The exponential growth rate of the annual traffic in the Internet pushes the need for a transport network that proves adequate for supporting this huge and varying amount of traffic, which by the way is almost exclusively based on IP protocol stack. The requirements of an efficient and cost-effective transport network for IP traffic are outlined in Chapter 1. The main requirements are scalability in terms of capacity, a good support of dynamic resource allocation and simple network management. Before we discuss the approaches for developing transport networks that meet these requirements, let us have a look at the evolution of transport networks until today.

[1] 1 PB = 10^{15} B

2.1 Transport Networks: Evolution Over The Time

The early transport networks, developed and standardized in late 1980s and early 1990s, were based on the Synchronous Optical Networking (SONET)/ Synchronous Digital Hierarchy (SDH) technology [31, 11, 32]. SONET/SDH is a synchronous TDM (time domain multiplexing) based circuit-switched approach that was introduced with the objective of providing a multiplexing layer for transporting digitized voice channels over high capacity optical fiber links. A voice channel encoded with the pulse code modulation (PCM) technique requires a fixed bandwidth of 64 Kb/s [33]. Several channels are multiplexed into a basic SONET/SDH frame and basic frames are hierarchically multiplexed into larger frames, which will be then transported over optical carriers. This approach for multiplexing a large number of fixed-rate digitized voice communications onto optical links proved to be very efficient over the years.

With introduction of data communication networks, the most straightforward approach to connect these networks together for exchange of data was to build an overlay network upon the network infrastructure developed for voice transport. In this approach, data communications should go through the voice circuits and via the voice channels through the SONET/SDH network. There were mainly two issues associated with this approach. First, data communications were packet based with variable data rates, whereas voice networks were based on fixed-capacity circuits (connection-oriented). This mismatch hinders efficient multiplexing of data communications onto the voice transport network. Consequently, solutions were proposed that were based on transporting data traffic directly over the SONET/SDH infrastructure and without using the overlay model upon the voice channels [34]. However, an adaptation layer was still needed to adapt the data bytes coming from data communication networks to the SONET/SDH . These efforts led to the deployment of the Frame Relay method for this purpose, see [35, 36], which was the successor of the packet-based network architecture that was designed for data communications and called X.25 [37]. Frame Relay is a packet-based approach that allows for multiplexing of data packets onto the SONET/SDH frames. The data unit in Frame Relay is called a frame and can be used to encapsulate several packets coming from voice and data protocols.

With further evolution of data communications networks, it turned out that there is a need for a QoS guarantee for data communications that was lacking in the Frame Relay approach. This contributed to the introduction of the asynchronous transfer mode (ATM) networking, see [38, 39]. ATM targeted to utilize the merits of both packet switching and circuit switching and to provide a generic networking technology for supporting efficient transport of both kinds of communications including data and voice. In ATM, packets are segmented into small fixed-size cells and the resulting cells are transported over a cell switching core that operates according to the asynchronous TDM. The ATM adopts the concept of virtual circuit (VC), which has to be established between two end points in the network before the exchange of data packets could take place. Many VCs share a physical link in the network and therefore statistical

multiplexing gains are achieved. Furthermore, the ATM technology comes with a suite of traffic shaping and admission control mechanisms that can be used to provision QoS guarantees. ATM was an approach that attracted lots of attention not only in the academia but also within network operators and networking equipment vendors [40, 41, 42, 43]. Today after some 20 years of its invention, a fraction of traffic in the Internet is still going through ATM switches. Nonetheless, the popularity of the ATM technology began to decline in early 2000s due to several factors. Some of these factors are the large overhead associated with having a header for each small cell (the so-called cell tax), complexity of the ATM protocol, the cost of segmentation and re-assembly of higher layer packets into and from ATM cells and last but not least lack of native ATM applications.

Following the developments in the ATM technology and identifying its shortcomings in late 1990s a new networking technology was introduced in 1997 named multiprotocol label switching (MPLS) [44, 45]. MPLS uses a switching technology that is to some extent similar to that of ATM. In MPLS a label is assigned to each data packet and the switching and routing of packets inside the networks are done based on their labels instead of their IP destination addresses. Furthermore, virtual paths – called label switched paths (LSP) – are established between two edge nodes in the network. The introduction of label switching and LSPs facilitates both the scalability of packet switching and the implementation of traffic engineering approaches in the network. MPLS is not a real layer-2 technology and is usually referred to as a layer 2.5 protocol as it sits between the layer 2 and layer 3 protocols in the protocol stack. Additionally MPLS can operate on any layer-2 technology including ATM. From early 2000s on with the convergence of almost all networking applications to IP, the IP-MPLS/ATM/SONET(SDH)/WDM became the established approach to transporting data traffic and as of today this is still valid to a large extent.

Over time, people started attempts to simplify the current multi-layer IP-MPLS/ ATM/SONET(SDH)/WDM approach. In fact, any of the layers participating in this approach owns an independent control plane. This makes provisioning of resources in the network a complicated and time-consuming task. Further, the developments in technologies deployed in any of the layers over the time have resulted in some cases to the existence of the same functionality in more than one layer, which is redundant. Consequently this leads to lack of efficiency and increase in OpEx. For the same reason, sometimes realizing the same functionality in one technology can be done with the lower cost than in others. For example, the recent advances in the optical technology has enabled the WDM layer to subsume some functionalities, like switching and multiplexing, that are usually realized in the adaptation layers above the WDM [6]. Also, the IP/MPLS can already perform fast provisioning of circuits over WDM networks. These are all driving forces behind simplifying the multi-layer transport network by eliminating the intermediate layers and transport IP/MPLS directly over WDM, i.e., the so-called IP and WDM integration [1, 46, 47, 48, 49]. Fig. 2.1 depicts the protocol stack before and after migration to transport networks based on IP over WDM.

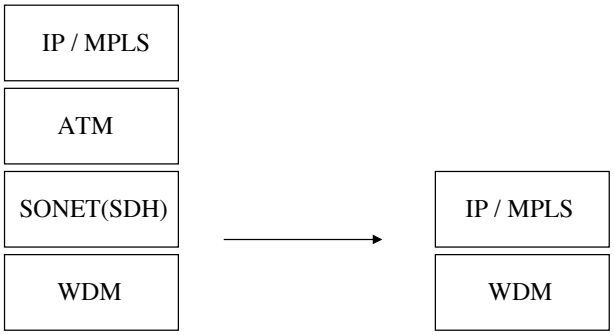

Figure 2.1: Migration from a multilayer transport network to IP over WDM.

2.2 IP/MPLS over WDM Techniques

Depending on the flexibility of the deployed WDM technology, there are different approaches to integrating IP over WDM. The major relevant approaches are reviewed and compared in this section.

2.2.1 Ultra High-speed IP Routers over Point-to-Point WDM

A straightforward approach to eliminating ATM and SONET/SDH from the IP transport architecture and increasing its efficiency is to connect ultra high-speed core IP routers via point-to-point WDM connections [47]. In this approach, optical transponders are required to convert the electrical (optical) signal at the output (input) interfaces of IP routers to the optical (electrical) signal. The optical signals will then go through add/drop multiplexers (ADM) and are transported over point-to-point WDM links as depicted in Fig. 2.2. In this approach, WDM is used merely as a physical layer technology and any kind of switching and multiplexing in the optical network has to be done through electrical cross connects. This necessitates the O/E/O conversion at the switching points and therefore the optical layer is not transparent.

Recently, major networking equipment vendors have developed IP routers with WDM interfaces, e.g. see [50]. That is, the optical transponder has been integrated to the line cards of IP routers, hence these new types of line cards are usually referred to as colored line cards. In addition, the recent developments in optical technology have made it possible to develop reconfigurable optical ADMs (ROADM) [6]. Combining the IP routers with the colored line cards and ROADMs to build IP over WDM networks brings several benefits over the old approach mentioned above and makes this approach more favorable. Specifically, with this arrangement there is no need to have O/E/O conversions at switching points in the WDM layer, since the electrical ADMs are replaced with the optical ones, which results in a lower CapEx. Also, the OpEx of the network is reduced through the improved manageability of the network.

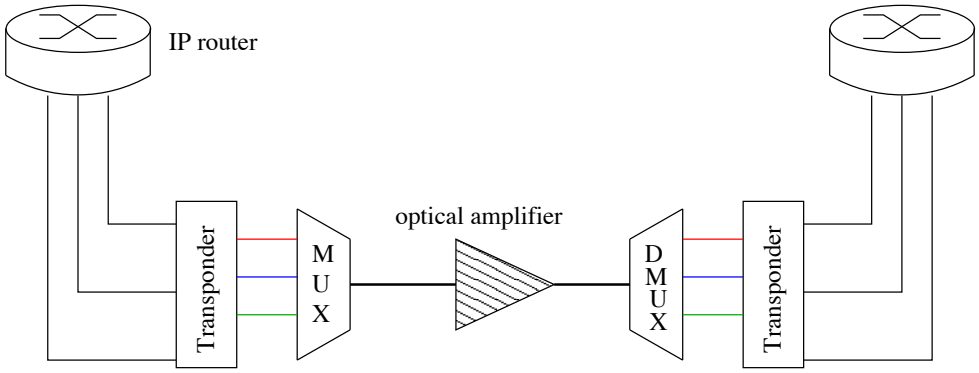

Figure 2.2: IP over point to point WDM links.

This technique eliminates the need for the multilayer transport as the traffic generated in the IP layer is directly transported over the WDM transport layer. This greatly improves the simplicity of setting up transport networks as well as their operation, administration and management (OAM). Moreover, this approach is fully based on IP and as a result it enjoys a high degree of flexibility and improved performance under bursty IP traffic. It is further amenable to the rich set of managements and control protocols that have been developed for IP.

Nevertheless, the exponential increase in the traffic generated in the Internet has brought concerns as to the scalability of this approach. The networking community has been discussing for several years now that electronic IP routers cannot keep pace with the increase in bandwidth demand and will soon become the bottleneck of the network. This argument is supported by the ever-increasing mismatch between growth rate of transmission speed of optical media and processing capacity of electronics.

On the other hand, major router vendors have been working intensively to utilize parallelism techniques to develop several-terabits IP routers. The cost of these ultra high-speed IP routers are however quite high, which makes this approach a costly one. In addition, the power consumption of the IP routers increase with the speed and it is expected that the ultra high-speed IP routers contribute significantly to the total power consumptions of ICT [51]. Optics offers a viable solution to all these issues. In fact, optics has achieved a maturity over the last several years that can take over some of the functionalities of IP in a cost-effective and less power-demanding fashion.

2.2.2 Wavelength Routed Optical Networks

Over the last 10-15 years, many sophisticated optical equipments have been developed that help the WDM layer to do some switching and routing functionalities and thereby healing the strain on the IP layer. One such device is a wavelength-routing switch that is usually known as an optical cross-connect (OXC). An OXC can in general switch

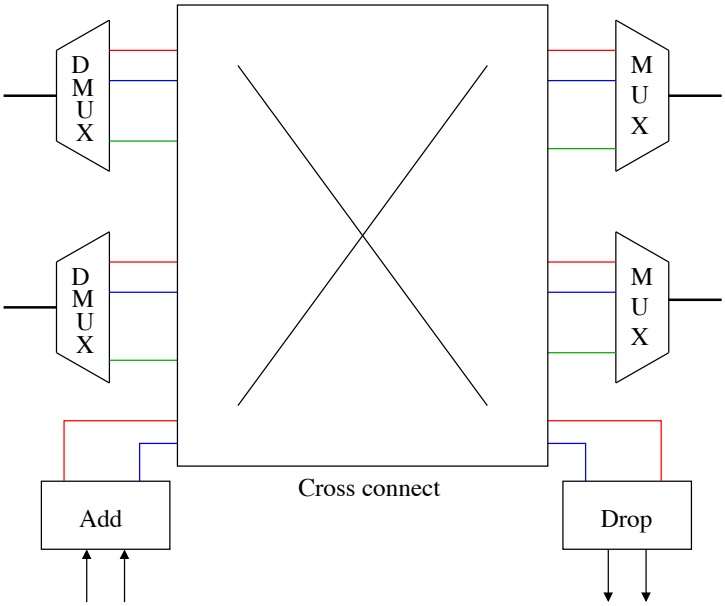

Figure 2.3: Schematic of a wavelength routing switch.

an optical WDM channel on any of its incoming fiber ports to any of its outgoing ports. To realize this functionality, an OXC is composed of optical multiplexers and demultiplexers and one (or sometimes more) cross connects as depicted in Fig. 2.3. The optical signal on any of the incoming fibers is first demultiplexed into the constituting wavelength channels and each channel can then be switched to any of the output ports independently. The configuration of the cross connects determines what channel from what incoming fiber is connected to what output fibers, which can be controlled electronically. The OXC allows for adding and/or removing data to/from any outgoing/incoming optical channel. That is, it also realizes the functionality of a reconfigurable ADM.

Making use of OXCs, the point-to-point WDM links can be replaced with a wavelength-routed optical network (WRON) [9, 10]. The WRON can then be utilized to embed a desired virtual topology on the physical optical network. An example is given in Fig. 2.4, where the traffic of an IP network with three nodes is carried over a transport network based on WRON. The connectivity required by the IP layer is provided by the optical layer through the virtual topology. The design of the virtual topology is carried out using the concept of lightpath [8]. A lightpath, by definition is *"an optical path (data channel) established between two nodes in the network, created by the allocation of the same wavelength throughout the path"*. Therefore, the problem of virtual topology design is to specify a set of all lightpaths that need to be established in the network in a way that some objective function is optimized. In the example of Fig. 2.4, two lightpaths are established in the network. The Lightpath A provides the connectivity

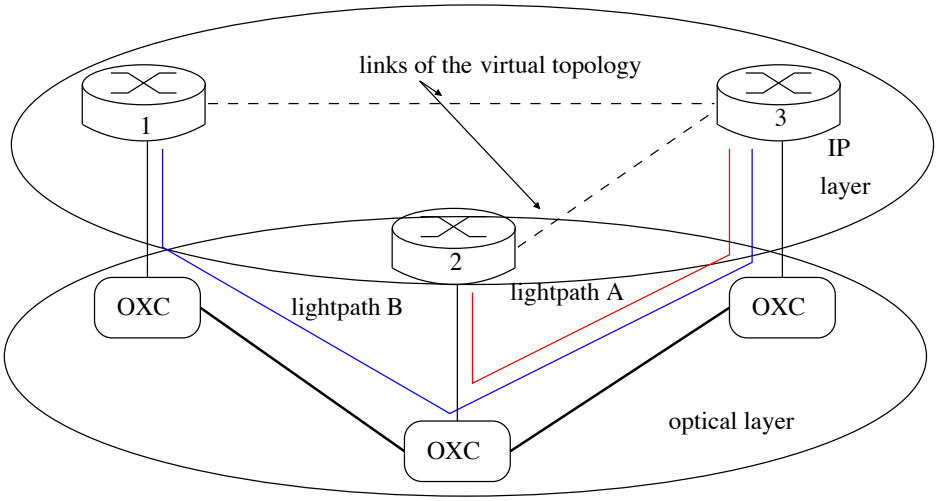

Figure 2.4: An example of WRON with 3 nodes.

between IP routers 2 and 3, and the lightpath B provides connectivity between nodes 1 and 3. The collection of the two lightpaths then form the virtual topology of the network.

Virtual topology design is one of the key building blocks of designing transport networks based on WRON approach, therefore it has been extensively investigated over the last several years [52, 53, 54, 11]. The virtual topology design can be formally stated as follows. Given are a physical optical network with certain set of networking resources – i.e. number of transponders available at each node as well as number of wavelength channels available on each link – and a traffic matrix. The traffic matrix specify the required bandwidth demand between each pair of nodes in the network. The goal is to find a virtual topology for the network subject to an objective function. Note that the set of nodes are the same for both physical and virtual topology as it is assumed that each node in the IP layer is associated with one and only one OXC in the optical layer. The problem can be solved subject to different objective functions [11]. A common practice for this purpose is to minimize the maximum congestion in the network, where the congestion is defined as the amount of load offered to a lightpath. In this case, the offered load to any lightpath in the network is minimized. Alternatively, one can solve the problem subject to minimization of average per packet delay across the network. Note that packet delay is composed of two major parts: propagation delay across the network, which depends on the routing as well as the length of the lightpaths, and times a packet spends in the queues of IP routers.

The virtual topology design problem with either of the objective functions explained above can be formulated as a mixed integer linear programming (MILP) optimization problem [53, 52]. The optimization problem, however, is NP-hard and therefore usually heuristic approaches are applied to solve it. To facilitate finding an approximate

solution to the problem it is usually decomposed into four subproblems as detailed below.

- Virtual Topology: Determine lightpaths that need to be established, that is between which two nodes in the network a lightpath is required.
- Lightpath Routing: Determine an appropriate route in the physical WDM network for each lightpath determined in the Virtual Topology subproblem.
- Wavelength Assignment: Assign a wavelength to each lightpath in a way that any wavelength channel on any link in the physical topology is assigned to at most one lightpath.
- Traffic Routing: route the IP packets between each source and destination pair over the designed virtual topology network.

The solution to the first subproblem determines the number of lightpaths and accordingly the number of optical transponders required in the network. The second and third subproblems are usually treated together and are therefore collectively referred to as routing and wavelength assignment (RWA) problem in the literature [55, 56, 57, 58]. The RWA problem has a direct impact on the number of wavelength channels required in the WDM links in the network. Finally, the fourth subproblem has an impact on the per packet delay of IP packets. Although the subproblems are not independent of each other, an approximate solution to the virtual topology problem can be found by solving each of the subproblems independently and putting the results together. Consequently, the resulting solution cannot be considered as an optimal one.

It has to be noted that solving any of the subproblems explained above is not an easy task for a large network. For instance, it has been shown that the wavelength assignment subproblem is NP-hard [59]. Therefore, approximation approaches have to be applied to solve the subproblems.

In the last several years many heuristic approaches to solving the virtual topology design problem have been proposed and investigated [10, 60, 52, 10]. In addition, the advances of the technology at some stages have contributed to solving the design problem. An instance of such cases is the introduction of wavelength converters [61], which can change the wavelength of optical channel. Having an OXC equipped with wavelength converters relaxes the constraint that a lightpath has to be routed over the same wavelength throughout the path and this, in turn, simplifies the wavelength assignment subproblem in the virtual topology design problem [57]. On the other hand, the introduction of wavelength converters to the WDM layer adds another possibility for optimization of the network. In fact, since the wavelength converters are expensive devices, OXCs can only be partially equipped with them or alternatively only some OXCs in the network should be equipped with converters. Consequently, an optimization here would be to determine how many converters are needed to achieve a given

performance in the network, or alternatively for a given number of available wavelength converters what OXCs are the best candidates for being equipped with wavelength converters – the so called converter placement problem in the network – in order that the network performance is maximized, e.g. [62, 63].

Transport networks based on WRON approach enjoy a high scalability, particularly when compared to the approach based on ultra high-speed IP routers. Nevertheless, they do not perform well when it comes to the efficiency and flexibility with respect to dynamic of IP traffic. In this context efficiency refers to the utilization of established lightpaths in the network, where each has a granularity of a single wavelength capacity (equal to 10 or 40 Gb/s in current WDM networks). In a real IP over WDM scenario, an edge IP router located at the entrance point of a lightpath can hardly fill this capacity and therefore lightpaths remain underutilized for most of their operational time. To improve this, traffic management techniques are applied to group traffic flows with small capacities into high capacity flows that can efficiently utilize the bandwidth offered by a lightpath. This process is usually referred to as traffic grooming. A large number of traffic grooming techniques have been developed over the past years [56, 64, 65, 66, 67]. The grooming techniques for IP over WDM networks can be in general divided into two categories: traffic grooming in the IP layer and traffic grooming in the transport layer.

Traffic grooming in the IP layer refers to the aggregation and multiplexing of traffic in the IP layer and can be embedded in traffic routing and virtual topology subproblems of the virtual topology design problem. Similarly, grooming in the transport layer corresponds to the aggregation and multiplexing of traffic from different edge nodes into lightpaths that is done in the optical layer. In order to do traffic grooming in the optical layer OXCs have to be replaced with the grooming OXCs, which are capable of multiplexing several incoming sub-wavelength traffic flows into one outgoing traffic flow. This is usually achieved with help of opaque OXCs, where an optical signal is converted to an electrical signal and again back to the optical after the multiplexing is accomplished in electronics.

Another drawback of the WRON-based transport networks is the lack of flexibility as mentioned above. In fact, a traffic pattern in the IP layer is not a static one, i.e., it changes over the time, which requires that lightpaths are established and torn down dynamically to reflect the dynamic requirements of the IP layer. This combined with the introduction of fast OXCs have contributed to the invention of dynamic wavelength routed optical networks (DWRON) [12, 13, 14]. In DWRON the IP layer can dynamically sent requests for establishing or deactivating lightpaths to the WDM layer. The control and interaction of IP and WDM layer in this approach can be done by employing the GMPLS control plane [68, 69].

2.2.3 Optical Packet Switching

Further developments of optical networking devices and the increase in the dynamic nature of data traffic in the IP layer in late 1990s shaped the interests in the networking community towards transport networks based on optical packets switching, which in turn resulted in launching several big projects on this topic across the world, e.g. [15, 18]. The approach was intensified over a short period of time in light of the following two facts. First, packet switching networks has proven quite successful in transporting highly dynamic traffic. The second fact has been increased concerns of the networking community over the scalability of electronic IP routers to terabits per second speeds.

In OPS, the ultra high-speed IP routers are replaced with optical packet switches, i.e., IP packets are converted to the optical domain at the edge of a network and are then switched all-optically inside the transport network [16, 17, 70, 71, 18, 72, 19]. The architectures proposed for OPS transport networks can be mainly categorized into synchronous switching (time-slotted) with fixed-size packets and asynchronous switching with fixed or variable size packets [6]. First let us consider the former one.

Fig. 2.5 depicts a generic architecture for a synchronous OPS core node, which mainly consists of optical demultiplexer and synchronizer units at input, optical regenerator and optical multiplexer units at output as well as an optical space switch, a control unit and possibly a contention resolution unit. Also depicted in Fig. 2.6 is the packet format in a switching time slot, which consists of header, payload and a guard time. The optical signal of each incoming fiber link is first demultiplexed into wavelength channels and then a small amount of signal power is tapped into the control unit of the switch that implements, among others, the header processing functionality. The control unit is usually realized in the electronics and therefore it has to be equipped with O/E/O conversion. The unit is further responsible for determining the configuration of the switch and accordingly commanding the switch in each time slot. The synchronizer unit at the input of the OPS node is mainly responsible for packet delineation and synchronization. In fact, although the operation of the network is synchronous, packets may arrive to the switches with different relative phases because of the chromatic dispersion in the fiber links, which leads to different propagation speeds for packets traveling over different wavelength channels on a fiber [6]. The synchronization unit can be realized through a combination of small size fiber delay loops. The regenerator unit at the output of the node, among others, takes care of 3R regeneration (reamplification, reshaping and retiming) of the optical signal, attachment of the new header generated by the control unit to the corresponding packet and resynchronizing the time slot. In case of packet contentions, they need to be resolved through the contention resolution unit. The possible ways of implementing contention resolution in an optical switch are discussed in detail in Chapter 3.

In the asynchronous OPS architecture there is no time-slotted switching and per packet switching of data is done in a continuous time fashion. In addition and in

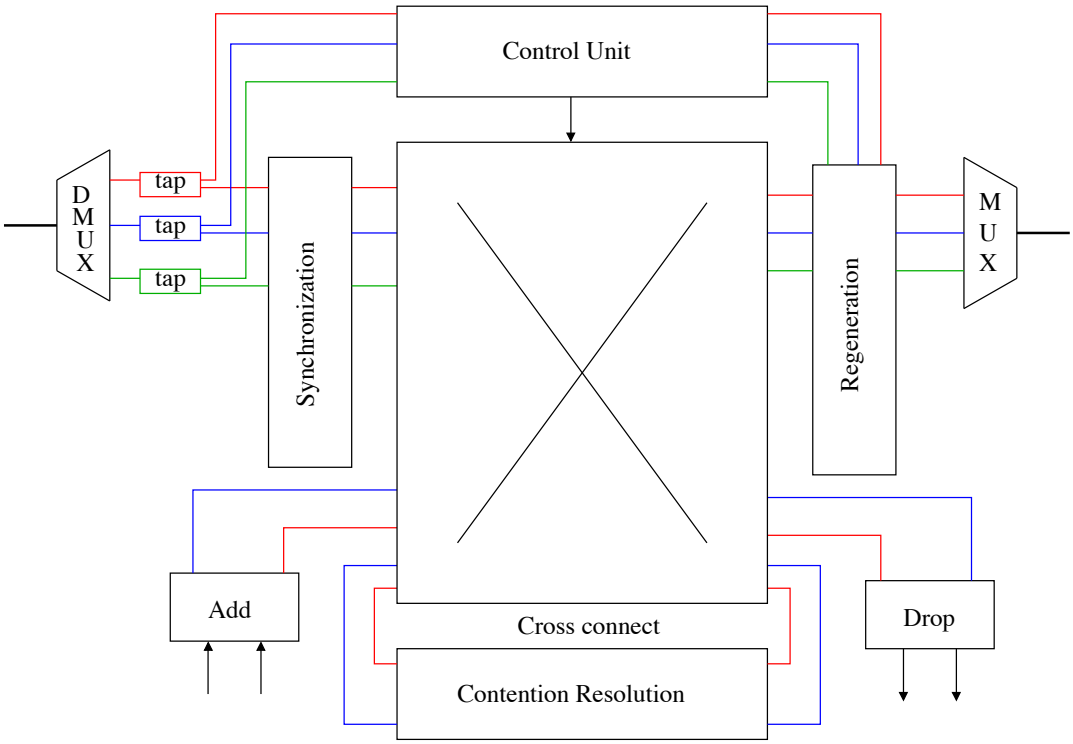

Figure 2.5: An abstract model of a synchronous OPS node.

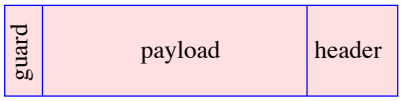

Figure 2.6: Packet format in a time slot.

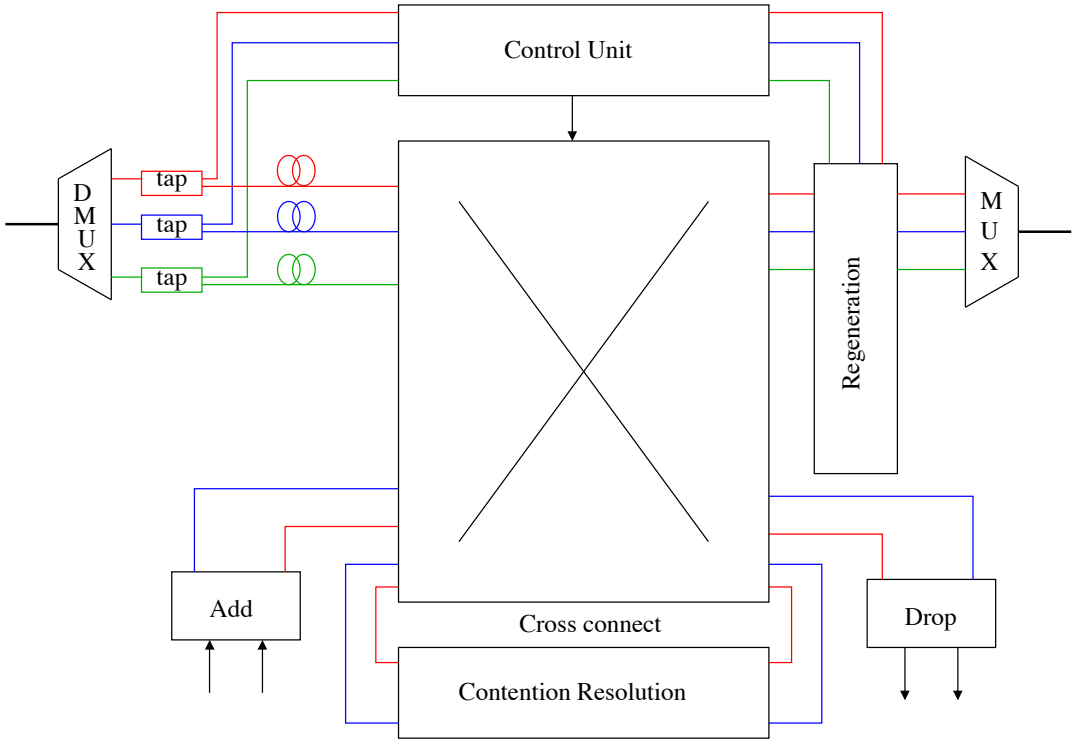

Figure 2.7: An abstract model of an asynchronous OPS node.

contrast to the synchronous mode of operation, there is no need for synchronization in the packet forwarding path of optical packets in an asynchronous OPS node, as shown in Fig. 2.7. In an asynchronous OPS node, the header recognition and extraction takes place in the control unit of the switch and the optical packets go through a small fixed-size fiber delay loop before being switched to the output of the switch. The role of the delay loops is to delay packets while the header extraction and processing are done, and consequently a forwarding decision is made in the control unit of the switch. Also, at the output of the node only header attachment and optical signal regeneration is needed as the need for packet resynchronization is eliminated in this architecture. It is intuitively understandable that under the asynchronous mode of operation the contention problem is more problematic than that under synchronous operation since in the former case packets can arrive to the node at any point in time.

Although OPS is a highly flexible approach for supporting dynamic traffic of the IP network, it suffers from several issues. First, in both synchronous and asynchronous optical packet switches described above, processing and routing of packets are done electronically, i.e., the approach is all-optical only in the forwarding plane. An immediate consequence of this is that the electronic-based control unit cannot scale well to much higher speeds. This has stimulated in the last couple of years research on the all-optical packet switching approaches, where the routing of packets are also done op-

tically. Unfortunately as of today, sophisticated all-optical processors cannot be built using current optical technology. Recently, all-optical gates and simple label swapping devices have been built in the laboratory [73, 74, 75], but they are still rudimentary and years of further research are required before they can be embedded in real systems.

The second major problem associated with OPS is the current mismatch between switching speed in the optical domain and optical transmission speed. To quantify this mismatch we note that over the last 25 years the switching capacity has increased by around three orders of magnitude, whereas the increase in transmission capacity over the same period has been more than four orders of magnitude [76].

Another issue associated with the OPS approach is the lack of optical RAM, which are central to any packet switching network architecture. In an OPS network buffers are required for two different purposes. First, buffers are needed to keep a packet while its header is being read and processed. Additionally, buffers are used to resolve contentions among different packets.

These three are the major issues hindering realization of OPS in the short-term, unless breakthroughs are made in the optical technology in the near future.

2.2.4 Optical Burst Switching

In late 1990s and early 2000s, a new approach termed optical burst switching was proposed for IP over WDM integration and received much attention in subsequent years [21, 20]. The approach follows the idea of burst switching, which was first introduced in the 1980s as a technique for fast transport of data and voice over TDM networks [77, 78]. Operation of OBS is very similar to that of packet switching, however it enjoys some unique features that makes its realization with state-of-the-art optical devices feasible. In this sense, OBS can be considered as an intermediate solution to the major limitations of optical communication technology that preclude the realization of OPS networks as explained in the previous section.

Specifically, OBS implements burst assembly at the network edge and out-of-band signaling in the network core. Burst assembly refers to the process of aggregating small-size packets into bursts at ingress edges of the optical network. By increasing the size of data units this aggregation makes it possible to relax the requirements on the speed of optical switching. Once a new data burst is ready for transmission at an ingress node, a signaling message is generated that contains all the required information for routing and switching of the data burst inside the network. Signaling messages are transmitted on dedicated WDM channels, i.e., out-of-band signaling. The benefit of such separation is that at intermediate nodes only signaling messages go through the O/E/O conversion – as they need to be processed electronically – and data bursts bypass such conversion. Switching of burst inside the OBS network can be carried out according to either store-and-forward or cut-through switching.

The OBS architecture relaxes the requirements on high-speed optical switches and all-optical processors. Nevertheless, it does not present any solid solution for replacing

buffers for contention control. As a result, a large number of research studies have been launched in the past few years on contention control in OBS networks. The operation and architecture of the OBS approach is discussed in more detail in Chapter 3.

2.2.5 Hybrid Optical Packet/Burst and WRON

Recently, there has been an increasing interest in new hybrid schemes that incorporate circuit switching (WRON) and packet/burst switching for transport of IP traffic, see [16, 79, 80]. The main objective of these new approaches is devising an optical transport network that enjoys the merits of both circuit and packet/burst switching while at the same time avoids their shortcomings. Gauger *et al* in [80] have classified the hybrid networks into three classes. Namely, a hybrid network can be realized in a client-server, parallel or integrated mode.

In a client-server hybrid network, a circuit-switched network acts as a server to the client packet/burst switched network. That is, the traffic of the OPS/OBS network will be transported over a WRON transport network. This is very similar to the high-speed optical router method, where a circuit switched network determines virtual topology of a higher layer network. The client-server hybrid network has the highest resource requirements and on other hand the lowest technology and control complexity compared to parallel and integrated hybrid networks.

Alternatively, in parallel hybrid optical networks, an edge IP router has access to both a WRON and an OBS/OPS network that operate in parallel to each other. In this method, based on some criteria such as QoS of the client layer traffic, the edge IP router has to decide which transport network should be used. In terms of resource requirements and complexity, this approach stands between the client-server and integrated hybrid networks.

Finally, in the third class of hybrid optical networks, there is only one transport network, whose resources are shared between the two switching technologies. That is, parts of the wavelength channels across the network are assigned to implement a circuit switching network and the rest will be used for the packet/burst switching. In comparison to the two hybrid networks explained above, this approach has the lowest resource requirements and the highest complexity.

The design of the integrated hybrid networks can be done aiming at either maximizing the network utilization or maximizing the quality of service of the transported IP layer traffic. There are three fundamental questions that need to be answered for this purpose. Starting from the extreme case of having all resources allocated to one switching paradigm, say OPS/OBS, the first question would be when and under what traffic conditions a circuit has to be set up in the network. The next important question in this regard is where exactly in the network, i.e., between which two nodes and over which path, this circuit has to be established. Finally, it is important to know what should the capacity of this circuit be. These three questions have remained unanswered in spite of the fact that the idea of hybrid networks is not a new topic. In fact,

a clear understanding of the performance and operation of this type of networks is only possible after answers to these questions are found.

2.3 Qualitative Comparison of IP-over-WDM Techniques

In the last section, all major solutions proposed for integrating IP with WDM were introduced and their corresponding features and capabilities were discussed. In this section, a unified qualitative comparison of the IP-over-WDM techniques is presented.

Shown in Table 2.1 is a comparison among five different IP-over-WDM techniques. Note that the Hybrid network considered here is of type parallel (see 2.2.5) as it lies between the other two methods in terms of resource requirements and complexity. The techniques are compared in terms of seven criteria that collectively show the merit of each technique. The merit of each technique with respect to any of the considered criteria is then marked with either "High", "Medium" or "Low".

It should be noted that performing a comparison among different IP-over-WDM techniques is a difficult and challenging task. One reason for such difficulty is that most of IP-over-WDM techniques are still conceptual frameworks. On the other hand, the fast technological advances over time can change the relative merit of one approach over others. What we present here is only a rough qualitative comparison that is deduced from comparing the general features of different approaches available in the literature.

First let us consider the cost of realizing each approach. We observe that OPS and ultra-high speed IP routers are the most costly approaches. These two are followed by OBS and Hybrid networks, while WRON is the best approach in this sense. The reason for the high cost of IP routers is related to issues such as complex parallelism that is required to achieve higher line card rates. As for the OPS, in fact, switching and processing of packets in the optical domain is quite expensive with the current technology, if possible at all. Therefore, we would benefit if processing is – at least partly – delegated to the electronics like in OBS and hybrid networks.

Another important quality metric to consider is flexibility, which here reflects the ability of a technique to adapt itself to the needs of the IP layer. In this sense, packet based approaches enjoy the highest flexibility and the WRON approach has the minimum flexibility among the considered approaches.

Support for QoS guarantee is another metric considered in Table 2.1. The major performance metrics considered here are delay, delay jitter and drop rate. It is known that QoS can be very well supported in circuit switched networks. However, it is very difficult to achieve a good degree of QoS support in packet switched networks, specially in terms of delay and delay jitter. These explain the differences in levels of QoS supports of different approaches in Table 2.1.

The ability to scale to higher capacities is another important criterion considered for comparison. Here, WRON and OPS outperform the others, owing to the fact

Table 2.1: Qualitative comparison of different IP-over-WDM techniques.

	UHS IP Router	WRON	OPS	OBS	Hybrid
Cost	High	Low	High	Medium	Medium
Flexibility	High	Low	High	High	Medium
QoS Support	Low	High	Low	Medium	Medium
Scalability	Low	High	Medium/High	Medium	Medium/High
Short-term Feasibility	High	High	Low	Medium	Medium
Statistical Multiplexing	High	Low	High	High	Medium
Network Resources' Utilization	High	Low	High	High	Medium

that they are not dependent on the electronic speed that is becoming a bottleneck in ultra high-speed networks. This also explains why electronics IP routers have a poor scalability. OBS and hybrid networks could be considered as moderately scalable solutions, because they still rely on electronics but not as much as IP routers do.

The short-term feasibility of an approach in Table 2.1 shows whether an approach can be realized with the state-of-the-art technology in the near future or not. IP routers and WRON are at the top of the list, since both are already commercially available. On the other extreme, there is OPS which is still many years away. Again, OBS and hybrid networks enjoy feasibility level that lies somewhere between OPS and WRON. Similar to the scalability metric, the factor that plays a major role here is how much an approach is dependent on the optical switching and optical processing power.

Table 2.1 further compares to what extent statistical multiplexing can be achieved in any of the approaches. It is seen that that ultra high-speed IP routers and OPS/OBS have the highest statistical multiplexing, while the WRON cannot achieve that much gain in this sense. The gain of the hybrid method lies between these two extreme cases. In general, as the resources are allocated in finer granularities – from circuit to burst to packet – the achievable statistical multiplexing gain in the network increases. Similar reasons hold true for the differences among the techniques in terms of network resources' utilization. In fact, utilization increases with the inverse of granularity of resource reservations, so that packet switching approaches, i.e., ultra high-speed IP routers and OPS/OBS can achieve the highest network utilization.

Chapter 3

Optical Burst Switching

In this chapter we first study architectural aspects and protocols of OBS and at the end define the system model that is considered for investigation in this dissertation. OBS is characterized by five building blocks: burst assembly, signaling and switching, channel reservation and contention control, see Fig. 3.1. These building blocks are detailed in the following sections. It is worth nothing that most of the system architectures and protocols discussed in this chapter are conceptual.

Fig. 3.2 depicts a generic architecture of an OBS node. There are two types of WDM channels multiplexed onto each fiber link: data channels and signaling channels. OBS adopts an out-of-band signaling method, where one or more WDM channels on each fiber link are dedicated to carrying signaling messages. The processing of signaling messages are delegated to an electronic node controller, which besides that, is responsible for making forwarding decisions, generating signaling messages for locally generated bursts, configuration of the optical switch as well as managing contention resolution resources.

The OBS network can use either offset-based or concurrent signaling. In the former one, the signaling message for a burst arrives to each OBS core node an offset time ahead of the burst to compensate for the time needed by the node controller to accomplish processing and forwarding procedures. In the latter one, however, both signaling and data messages arrive at the switch at the same time. In this case, the input unit of the node will take care of optically delaying data messages while the corresponding control messages are being processed. Also, the output unit of the node has the responsibility to compensate the impairments of the optical signal.

The burst assembly unit aggregates locally generated traffic to the bursts that are suitable for transmission in the OBS network. Also the disassembly unit fragments bursts destined to the node into constituting data packets. Contention resolution is a crucial functionality of an OBS node and as will be seen later in this chapter, it can be implemented in a variety of ways. For example, the contention resolution resources can be placed in an input unit or output unit, or as shown in Fig. 3.2, they can be utilized in a central way in the node. In the rest of this chapter, the main building

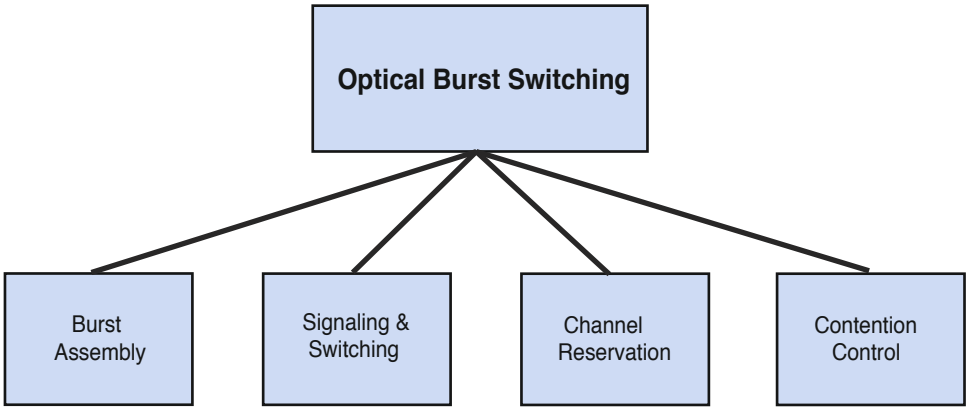

Figure 3.1: Building blocks of optical burst switching.

blocks of the OBS architecture are studied in more detail.

3.1 Burst Assembly

The rather slow growth rate of optical switching speeds has brought about a mismatch between transmission speed of optical links and switching speed of optical devices. As a result, directly switching IP packets in the optical domain, can introduce very large switching overheads. For the sake of illustration let us consider an example. It is known that around 40% of packets in the Internet are only 40 B long, which corresponds to TCP acknowledgment packets [81, 82]. The amount of time needed for transmitting a packet as small as 40 B over an optical channel operating at 40 Gb/s is equal to 8 ns. That is, to achieve a reasonable switching performance the switching speed needs to be much smaller than 8 ns. Such fast optical OXCs are not commercially available yet. In fact, optical switches with different switching speeds are built based on different technologies. The fastest types of optical switches are acousto-optic, electro-optic and semiconductor optical amplifier (SOA) based switches, with switching speeds in the range of 3 μs, 10 ns and 1 ns, respectively [83]. Therefore, a natural solution to this issue is to generate jumbo packets – bursts – by means of aggregating small-size packets. Design and implementation of a properly tuned burst aggregation mechanism is indispensable to the realization of any OBS network.

In order to generate bursts, each OBS ingress node contains a burst assembly unit as depicted in Fig. 3.3. The unit receives packet-level traffic, which is usually collected from low speed links at its input and aggregate them into bursts. In its simplest form, an assembly unit is composed of a packet classifier, an assembly controller and several assembly buffers as shown in Fig. 3.3. Once a packet arrives to the unit, its destination address (and possibly the QoS class that it belongs to) is checked by the packet classifier. According to result of the classification, the packet is enqueued in

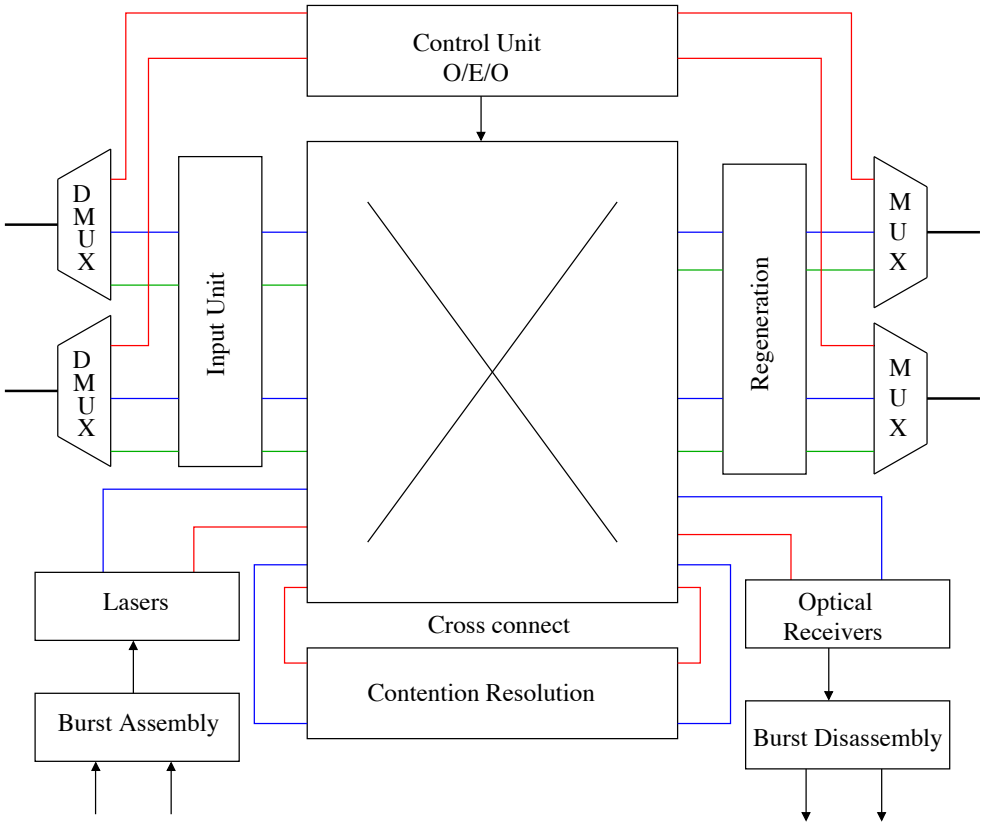

Figure 3.2: An abstract model of an OBS node.

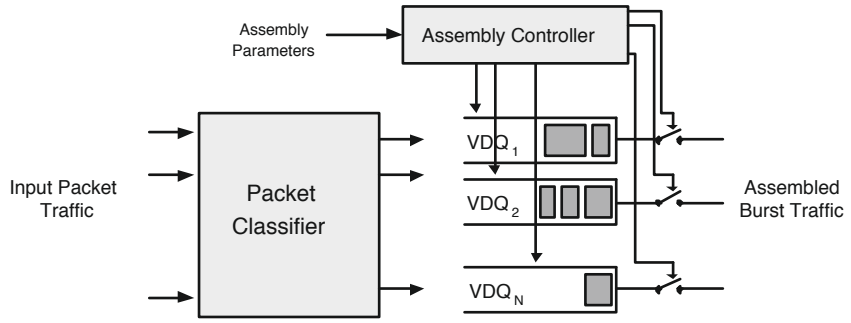

Figure 3.3: Burst assembly unit in an ingress OBS node.

one of the available virtual destination queues (VDQs), where each VDQ is a buffer dedicated to packets destined to a certain egress node, and is referred to as an assembly buffer. Therefore, the unit should contain one assembly buffer per destination (and per QoS class). Note that the term destination here refers to an egress node in the OBS cloud, thus packets of the same attributes from different micro flows can be multiplexed into the same assembly buffer.

The assembly controller is responsible for making the decisions regarding when contents of each assembly buffer should be aggregated into a burst and released to the network. This controller can be also incorporated into the control unit of the OBS node, see Fig. 3.2. The controller takes care of scheduling burst generations based on several criteria, of which some are imposed by the network, e.g., maximum and minimum burst length, and others are imposed by QoS requirements of incoming packet traffic, e.g., maximum delay that a packet can tolerate in an assembly buffer. Accordingly, various proposals have been presented and investigated for this purpose, see [84, 85, 86, 87, 88, 89, 90]. The controller may apply different algorithms to different assembly buffers.

In general, burst assembly algorithms may be classified into three major categories, namely volume-based, time-based and hybrid algorithms. Principles of all three algorithms are discussed in the following.

3.1.1 Time-based Burst Assembly

In a time-based algorithm, the controller is equipped with a timer. Once a packet arrives to an empty assembly buffer, the timer associated to the corresponding buffer is set to a time threshold T_{Th}. This threshold is determined considering maximum delay that a packet can tolerate in the ingress node. Then, as soon as the timer expires, all packets in the buffer are aggregated into a burst and sent out. The timer is deactivated when the buffer is emptied. If length of the burst generated in this way is below a given level L_{min}, padding has to be used to fulfill the minimum burst length requirements. The value of L_{min} is dictated by the network architecture [84], and

depends on the ratio between number of data and control channels of WDM links in the network. Specifically, L_{min} has to be selected large enough so as to avoid possible conflicts between reservation messages of different data bursts over the control channel. The reason why we need to set L_{min} will be explained in more detail in next section.

3.1.2 Volume-based Burst Assembly

In a volume-based algorithm, the controller checks the aggregate length of packets in the assembly buffer each time a new packet arrives. As soon an the aggregate length exceeds a predefined threshold L_{Th}, all packets in the buffer are assembled into a new burst. This threshold should clearly be larger than the minimum burst length requirement of the network and is usually selected with respect to maximization of utilization of resources over the network [84]. Algorithms of this category are usually not recommended for delay-sensitive traffic, because there is no guarantee on the upper-bound delay that a packet may experience in the buffer; in fact per packet delay in this case will be in direct relation with the offered load to the assembly buffer.

3.1.3 Hybrid Burst Assembly

In a hybrid assembly algorithm, the control unit keeps track of both aggregate volume of packets in the buffer and the time elapsed since the first packet has arrived. That is, the timer is set to T_{Th} once a packet arrives and finds the buffer empty, and length of the queue is compared against a length threshold L_{Th} upon each new arrival. Then, a new burst will be generated when either the timer expires or the volume threshold is exceeded. In either case, the timer will be deactivated after the buffer is emptied. In an algorithm of this category, load intensity determines which criterion, between time and volume, has to be used to generate a new burst at a given time. That is, if the load intensity is below a specific level, a new burst will be generated T_{Th} units of time after the first packet has arrived; however, if the load intensity is heavy enough, bursts of length L_{Th} will be generated back to back so that no packet will encounter maximum assembly delay of T_{Th}. In the former case, it is likely that the timer expires while the aggregate length is less than the minimum burst length requirement. If this happens, padding has to be used. Hybrid burst assembly is in fact the most general algorithm among the three categories, since it also encompasses the other two categories.

After a burst has been generated by the burst assembly unit, it will be queued in a burst queue. Then, bursts in the burst queue will be processed by the control unit and released to the network. How bursts are transmitted through the OBS network will be discussed in the following section.

3.2 Signaling and Switching

In the last section we observed how burst aggregation mechanism can resolve the problem associated with slow switching speed in the optical domain. There are still two issues that need to be addressed, namely the issue related to lack of mature all-optical processors and that of contentions among bursts. In fact, one of the major challenges of implementing a packet-based optical transport network is how to process packets at every core node of the network without using O/E/O conversion. On the other hand this cannot be done efficiently in the optical domain because all-optical processors are still rudimentary and therefore cannot be used for this purpose. The OBS paradigm addresses this issue by introducing an out-of-band signaling mechanism that makes it possible to use powerful electronic processors to process headers of optical bursts without converting the data unit into the electrical domain. In the following, we first consider out-of-band signaling of bursts and then turn to the switching of bursts in the core of an OBS network.

3.2.1 Out-of-band Signaling

In the OBS paradigm, the problem related to efficient processing of burst headers is addressed through decoupling the header of a data burst from its payload and transmitting each part separately. This technique is known as *out-of-band* signaling and has the advantage that only burst headers traveling over dedicated signaling channels have to go through O/E/O conversion in intermediate nodes [21]. In other words, using this mechanism the payload of the burst, which is orders of magnitude larger in size than its corresponding header, will remain in the optical domain all along its path inside the OBS network.

In a network with out-of-band signaling, k out of W wavelength channels on every optical link of the network are dedicated to control packets and the remaining $W - k$ to the data bursts. That is, data bursts and the corresponding headers – also known as burst header packet (BHP) – are transmitted separately over dedicated wavelength channels on the same link.

Taking into account that the ratio of the average size of data bursts to that of BHP is quite large, OBS networks can be designed with the number of signaling channels being significantly smaller than that of data channels. Moreover, signaling channels can operate at a much lower data transmission rate than data channels. This, in turn, relaxes the requirements on optical signal receiver, O/E/O converter and the packet processor in intermediate network nodes. This implies that flexible processors based on mature electronic technology can be employed in realizing the control unit of OBS core nodes.

Since only a small fraction of WDM channels on any link is dedicated to the signaling, say one channel on a link with total 16 channels [84], precaution should be taken to avoid congestion on the signaling channels. For example consider a design example

where only one out of ten available WDM channels is devoted to signaling and the rest is used for data bursts, i.e., the ratio of number of data channels to that of signaling is nine. In this case there is a high chance of congestion on the signaling channel if the average burst size is smaller than 9 times the average BHP size. Following the framework presented in [84], for a given partitioning of channels on WDM links in the network, there should be a limit on the minimum acceptable data burst size. More specifically, suppose each link in a given OBS network has W wavelength channels in total, out of which $W - k$ are data channels and k are signaling channels ($k < W$). Also, let us assume the average size of data bursts and BHPs in the network be B and b ($b > B$) seconds, respectively. To avoid congestion on the signaling channels in a highly loaded link, the average burst transmission rate $(W - k)/B$ has to be equal to or smaller than the average BHP transmission rate k/b. That is,

$$B \geq (W - k)b/k. \tag{3.1}$$

3.2.2 Switching of Bursts inside the Network

Let us now consider how a burst is switched in a node of an OBS network. Once a signaling message arrives in an intermediate node, it will take some time until the message is fully processed by the node control unit and the switch matrix is configured to switch the associated data burst from an input data channel to a channel on the right output link. There are two approaches to implement the switching depending on what happens on the data unit during this time: store and forward and cut through switching. In the following these two techniques are introduced and their implementation in the OBS context are discussed.

Store and Forward This is the technique that is usually used in electronic packet-switched networks. To allow for the header processing and switch configuration in this technique, packets are temporarily stored in buffers and remain there until the control unit of the node reads and analyzes header of the packets and determines the output port that each packet has to be forwarded to, hence the name store and forward [91]. RAM memories are central to this approach, however, as pointed out before there is no equivalent to RAMs in the optical domain. Therefore and in order to implement this approach in OBS networks fiber delay lines are used instead of RAM buffers. Specifically, a short FDL is employed at the input unit of each node in the path of data channels. In this setting, both data burst and signaling message (BHP) are released to the network at the same time by the ingress node and therefore they arrive at a core node almost simultaneously. Inside a core node, the BHP enters the control unit of the node and the corresponding data burst enters a delay loop in the input unit. The length of the delay loop must be selected in a way that the burst starts to leave the loop just after the switch has been properly configured to forward the burst to a proper output port. At the same time, the signaling packet is updated and forwarded over

the signaling channel of the output link over which the data burst has been forwarded to. That is, the data burst and associated BHP leaves the node at the same time.

Cut-through Switching An alternative to the store-and-forward switching paradigm is cut-through switching, where a data burst cuts through the switch as soon as it arrives at a core node. That is, forwarding decision and switch configuration for every single burst have to be made during the header arrival. A signaling method known as *offset-based signaling* enables such a cut through switching inside OBS [21].

Using the offset-based signaling the processing and switching times for every signaling message are compensated at the edge of the OBS network, where large electronic buffers are available. The operation of an OBS ingress node in a cut-through switching network is as follows. Once a new burst is generated by the burst assembly unit and it is ready for injection into the network, a signaling message (BHP) is generated by the control unit of the node. The signaling message is released to the network on a control channel of an appropriate output link an *offset time* ahead of the burst itself. In other words, transmission of the data burst will be delayed by the ingress node, hence the name offset-based signaling. At the same time the reservation of the resources must be made by the ingress node to guarantee the transmission of the data burst at the scheduled time. The generated BHP must include all information required for successful switching of corresponding data burst along the path to its destination. Among others, it must contain the exact amount of offset time to allow the intermediate nodes to synchronize the BHP with the arrival time of the corresponding data burst. More precisely, each intermediate node can calculate the arrival time of a data burst by adding the offset time to the arrival time of the BHP to the node. Also, each intermediate node must update the offset time in each BHP before forwarding it to the next node. That is, it must reduce the offset time by the amount of time the BHP has spent in that node.

The offset time is selected large enough so that it makes up for all processing and switching times the associated signaling message may encounter during its journey inside the network, as shown in example scenario of Fig. 3.4. More specifically, let us assume that processing and switching times are fixed in all nodes in an OBS network and are equal to t_p and t_s, respectively. In this case, if a specific data burst is supposed to be switched though k nodes in the OBS network, the offset time between the data burst and its associated BHP must be set to $k \times t_p + t_s + t_g$, where t_g is a guard time. This will guarantee that all the k nodes in the path of the burst will be configured before the data burst arrives and accordingly the data burst can cut through the switches [21]. Note that in calculating the required offset time it is assumed that as as soon as a signaling message is processed by a core node, it can be forwarded to the next node. That is, the switch configuration can be done in parallel to the forwarding of BHP packet to next hop.

The cut-through switching based on offset-based signaling simplifies the optical forwarding plane of the OBS network through eliminating the need for buffering data

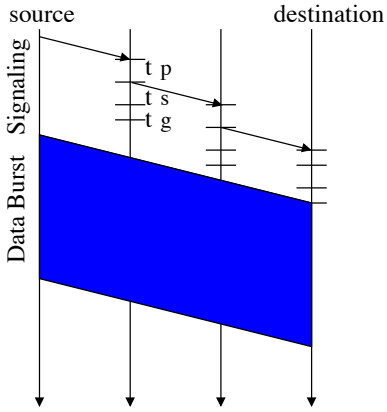

Figure 3.4: Offset-time based signaling.

packets during header processing and switch configuration. This explains how another technological barrier to implement optical packet-switched transport networks is circumvented. Nonetheless, there are also some drawbacks associated with this technique. First and foremost the offset-based signaling gives rise to unfairness among packets with different offset times in accessing networking resources in the core of network. In fact, as detailed in the next section, channel reservation for every individual data burst is made in advance – depending the amount of offset time associated with a reservation request – upon the arrival of its associated BHP. On the other hand, as shown in Fig. 3.4, the offset time between a BHP and its corresponding data bursts decreases at every hop along the path so that at the last hop it becomes very small and equal to $t_p + t_s + t_g$. This implies that any core node may receive reservation requests with different offset times, where the offset of any request depends on the number of further hops a data burst has to travel. In this case, those reservations that have a larger offset time, i.e, are further away from their destinations, will be assigned resources with higher probabilities than those with smaller offset times. This can be illustrated by a real life example. One is more likely to successfully reserve a ticket for a popular performance if one applies quite in advance. That is, the probability that one can obtain a ticket is in direct relation with how early one applies for it. This is known as the fairness problem of offset-based signaling and has stimulated several research works to overcome it.

Although the fairness issue is a drawback of using offset-based signaling, it opens up a new direction for implementing QoS differentiation in OBS networks. Namely, it is proposed to utilize the effect of having different offset times to provide service differentiation to bursts of different classes [92, 93, 94, 95]. In this approach, the burst traffic is first classified into two classes: high priority and low priority. For the bursts belonging to the high priority class, the offset time is intentionally increased beyond the originally required offset time. This can increase success probability of burst headers

in reserving channels in the core of the network. Authors in [94] show that a good isolation among different classes of traffic can be achieved by using the offset time technique.

Another drawback of the offset-based signaling is that it requires the path of each burst in the network to be fixed and known in advance by the ingress edge node. Consequently, some contention resolution techniques that are based on alternative routings of bursts in times of contention in the network, e.g., deflection routing, cannot be easily applied to the OBS networks that operate based on offset-time signaling.

3.3 Channel Reservation Algorithms

When a BHP arrives at an OBS core node, it is processed by the control unit and the resources are reserved for the associated burst accordingly. This has to be done through a channel reservation algorithm. Several channel reservation algorithms with different levels of complexity and performance have been proposed for OBS networks [96, 97, 84, 98, 99, 100, 101, 102]. Following the classification presented in [103] there are two general classes of channel reservation algorithms, namely, algorithms based on immediate or delayed reservations. In the algorithms of the former class, once a core node receives a BHP that is associated with a data burst to arrive on an incoming fiber link on a given wavelength channel, it reserves the same wavelength channel on an appropriate outgoing fiber link *immediately* if the channel is not busy. However, in the algorithms belonging to the latter class, the reservation is done for a *later time* associated with the estimated arrival time of the data burst. It should be noted that the difference between immediate and delayed reservation schemes is relevant only in networks employing offset-time-based signaling method. In other words, both the classes perform similarly under store-and-forward signaling method.

There are a couple of algorithms that fall into either of these two categories. Three salient such algorithms are just in time (JIT), Horizon and just enough time (JET), where JIT is an immediate reservation method and the two others are delayed based. In the following we look into these three specific algorithms in more details. In explaining different channel reservation algorithms, an ideal synchronization is assumed between BHPs and corresponding data bursts.

3.3.1 Just in Time

Just in time is among the simplest channel reservation algorithms and follows the principles of immediate reservation [97]. That is, the control unit reserves the appropriate output channel for a data burst upon the arrival of its signaling message and in case the desired output channel is not idle at that time, the reservation fails. In other words, the output channel will be reserved for a burst from the time a BHP is processed by the core node, thus under the offset-based signaling technique, the output channel will

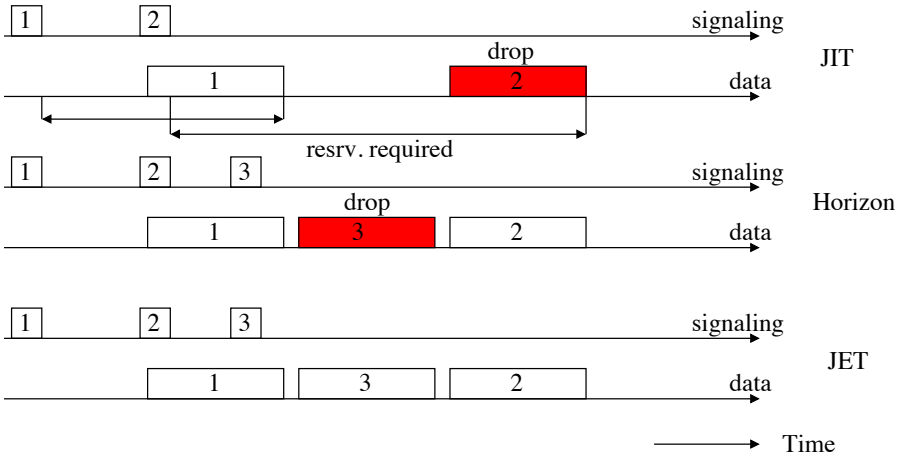

Figure 3.5: A comparison among JIT, Horizon and JET channel reservation schemes.

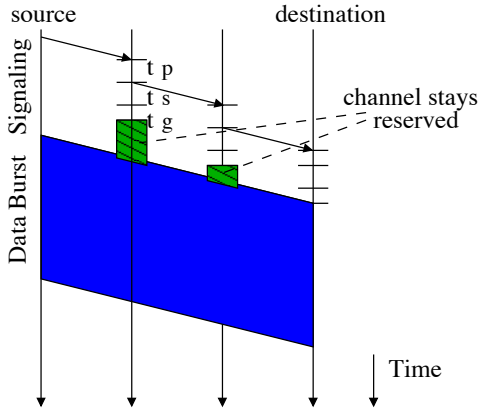

Figure 3.6: Just in time channel reservation.

remain idle during the offset period. The example of Fig. 3.6 schematically shows the timing of channel reservation using the JIT scheme. Also, the diagram in Fig. 3.5 shows an example of the operation of JIT in a system with one signaling channel and one data channel. In this example, the marked areas show the periods over which the channel remains reserved for associated bursts. It is seen that although burst 2 can be accommodated over the channel, it is blocked since at the time BHP 2 arrives to the system the data channel is still busy.

In terms of implementation complexity JIT is one of the simplest channel reservation schemes. In fact, a channel scheduler that implements JIT only needs to know at any time if a data channel is idle or not. Despite its simplicity, JIT suffers from poor performance, in particular for cases where the offset time is large and comparable to the size of data bursts. This can result in a high burst blocking rate.

3.3.2 Horizon

The issue of JIT reservation scheme is that a channel is reserved for the total duration of offset time plus burst length as it was illustrated in example of Fig. 3.5. This issue has been addressed in the delayed reservation schemes. A simple such scheme is Horizon, in which a channel will be reserved for a limited duration equal to the exact length of a burst [20]. To implement the Horizon, the scheduler will need to know the time that the burst arrives to the switch as well as the burst duration. Also, at any time the scheduler has to keep track of the time after which the channel has not been scheduled and is idle, i.e., channel horizon. Accordingly, when a new reservation request arrives, if the burst arrival time is after the current channel horizon, the reservation will be made successfully. The channel horizon will be then updated to the time at which the arriving burst will completely leave the channel. However, if the burst arrival time is earlier than the channel horizon, the reservation will fail. An example of using Horizon is also depicted in Fig. 3.5. In this example, burst 2 that would have been blocked by the JIT algorithm is accepted by the system. However, burst 3 will be blocked since at the time BHP 3 arrives at the system, the channel horizon is equal to the departure time of burst 2.

3.3.3 Just Enough Time

Just enough time is another reservation scheme that, similar to Horizon, belongs to the delayed reservation algorithm. It tries to improve the performance of Horizon by utilizing the gaps that are created over the channel due to the existence of an offset time between a data burst and its corresponding BHP [96]. In fact, the channel will remain idle during this offset time, which can lead to performance degradation. For instance in the example given in Fig. 3.5, burst 3 will be blocked under Horizon despite the channel being idle for the duration of the burst. In the same example, burst 3 will be accepted in the system if JET is used. In JET, when a new reservation message is received, first the exact start t_s and end time t_e of a corresponding burst is computed. Then, the scheduler checks if the channel is idle during the time interval $[t_s, t_e]$. If so, the burst will be accepted, otherwise the reservation fails. It is worth to mention that in contrast to JIT and Horizon, JET leads to a non-FIFO (first in first out) service discipline of bursts with respect to the arrival times of their associated BHPs.

In comparison to JIT and Horizon, JET can achieve a better performance since it has the potential to utilize all gaps over the channel. In fact, in JET the network resources are utilized more efficiently. Nevertheless, the use of JET necessitates much more computational capabilities at core nodes, since the scheduler has to keep track of all time intervals, during which the channel is scheduled.

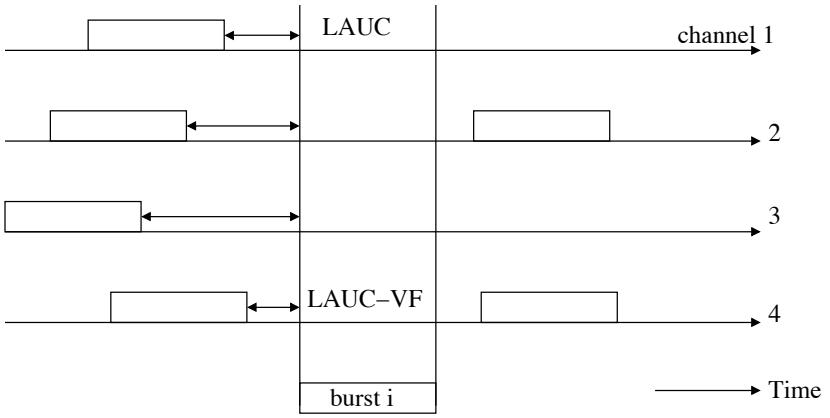

Figure 3.7: Comparison between LAUC and LAUC-VF. LAUC schedules burst i on channel 1, while LAUC-VF schedules it on channel 4.

3.3.4 Channel Reservation In The Presence of Wavelength Converters

Upon the arrival of a burst to an OBS node consisting of WDM channels and wavelength converters, it might happen that the scheduler finds more than one channel which are able to serve the burst, i.e., the case that more than one eligible channel exist. Depending on how to select one of the eligible channels under such situations, different variations of the above-described reservation algorithms can be developed. The most sophisticated algorithms of this type are latest available unscheduled channel (LAUC), and LAUC with void filling (LAUC-VF), which are variations of Horizon and JET, respectively [84]. In LAUC every new burst is scheduled over one of the eligible channels in a way that the gap between the newly accepted burst and the burst before that is minimized. That is, LAUC tries to minimize the fragmentation of channels. LAUC-VF is the same as LAUC except that it also supports void filling. An example is given in Fig. 3.7, where burst i can be accommodated on any of the four channels of the WDM link. If LAUC is used to schedule the burst, only channels 1 and 3 will be eligible; however, the gap created over the channel is smaller if channel 1 is chosen. As a result the burst will be served on channel 1 in LAUC. In LAUC-VF all the four channels are eligible and channel 4 is chosen since it minimizes the gap.

3.4 Contention Control

Since bursts from different source nodes arrive at a core node asynchronously, contentions may occur when two or more data bursts want to use the same output channel of a node at the same time. If no mechanism is used to address the contention problem, all contending bursts except the one that has arrived first have to be dropped,

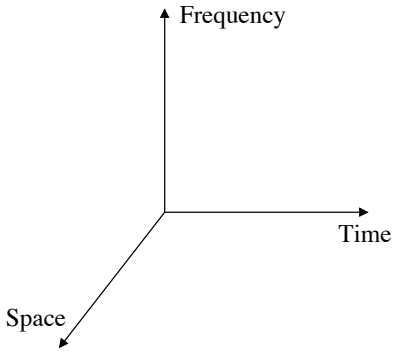

Figure 3.8: Basic elements of contention control in optical domain.

which can result in an unacceptable performance of the node and the whole network. Taking into account the fact that each data burst in the OBS network is the result of aggregating several IP packets, appropriately controlling the contentions is a crucial issue. Most of the approaches to controlling the burst contentions are based on reactive methods. That is, they try to resolve the problem only after the contentions occur. Therefore, these techniques are usually referred to in the literature as the contention resolution methods.

In general, there are three basic directions that form the playground for resolving contention in optical networks as depicted in Fig. 3.8: frequency, time and space, which are respectively represented by wavelength conversion, FDL buffering and deflection routing [104]. A contention resolution mechanism can incorporate only one of these basic techniques or can alternatively apply a hybrid (combinational) scheme by employing two or three of them. Fig. 3.9 depicts taxonomy of contention resolution schemes in OBS networks. In the following we discuss each of the mechanisms in more detail.

3.4.1 Wavelength Conversion

According to the famous $Erlang_B$ formulae in queueing theory it is known that in a loss system with a given offered load per server the probability of call blocking can be effectively reduced by increasing the number of servers [105]. In fact, this improvement stems from the statistical multiplexing gain that increases with the number of shared servers. This is the main idea of using wavelength converters in WDM optical networks. Consider a situation that a burst arriving on a given wavelength channel is blocked because the channel with the same wavelength on the appropriate output link is busy. In this situation, a wavelength converter can be used to transmit the arriving burst on an idle channel of the considered output link on a wavelength different from its original one. In other words, the use of wavelength converters enables inter-wavelength statistical multiplexing. There are two types of wavelength converters: fixed and tunable

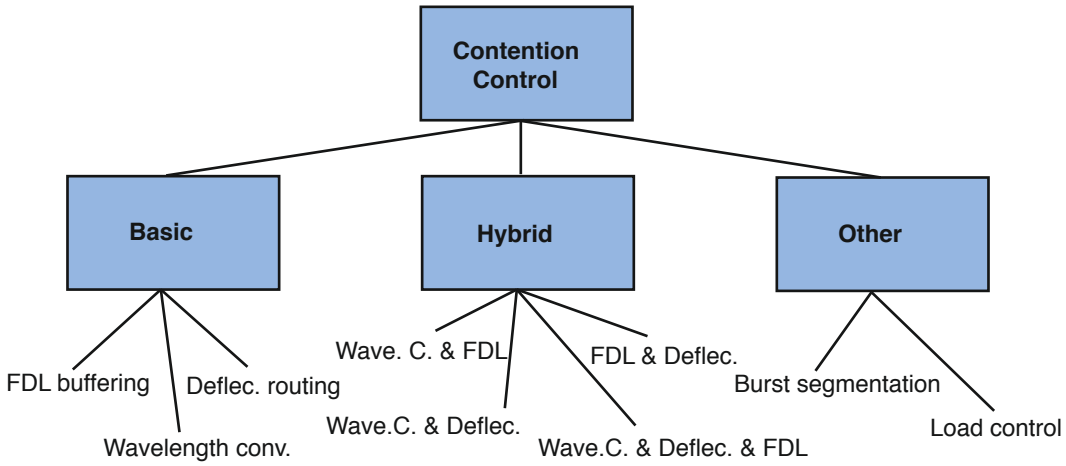

Figure 3.9: Taxonomy of classical contention control mechanisms.

[106, 107, 108, 61, 6]. In fixed wavelength converters, wavelength at the input or/and output of the converter is always fixed. In contrast to fixed converters, the tunable ones have the ability to convert any incoming wavelength to any outgoing one. There are also wavelength converters that have a limited wavelength conversion range, that is, an incoming wavelength can be converted only to one of the adjacent wavelengths.

Wavelength converters can be employed in a core switching node in different ways [61, 109]. In the simplest form, which is known as *full wavelength conversion*, one converter can be used per input (output) wavelength channel. Figs. 3.10 and 3.11 depict a switch architecture with full wavelength conversion at input and at output, respectively. In this case, wavelength converters of type fixed can be used. In a switch equipped with full wavelength conversion, a burst arriving to the switch on any wavelength can be transmitted on any wavelength channel of desired output port only if that channel is idle. If the number of wavelength channels per port is large, full wavelength conversion can greatly reduce the loss rate of the system, however, this might not be cost optimized, since a wavelength converter is an expensive element.

Alternatively, it is possible to equip an OBS switch to the conversion capability only partially, i.e. *partial wavelength conversion* [110, 111]. In implementing an OBS switch with partial wavelength conversion, another degree of freedom is introduced, which is how to integrate the converters into the switch. One possibility here is to create a pool of converters that is shared among all ports of the switch, i.e., shared per node (*SPN*) [61] as illustrated in Fig. 3.12. Alternatively, one can create a dedicated pool of converters at each output port (link) of the switch [61]. In this case, wavelength converters are referred to as shared per link (*SPL*). An example of the switch architecture with SPL wavelength conversion is shown in Fig. 3.13.

In comparison to the full wavelength conversion, the partial wavelength conversion technique has the advantage that a smaller number of converters is required in order

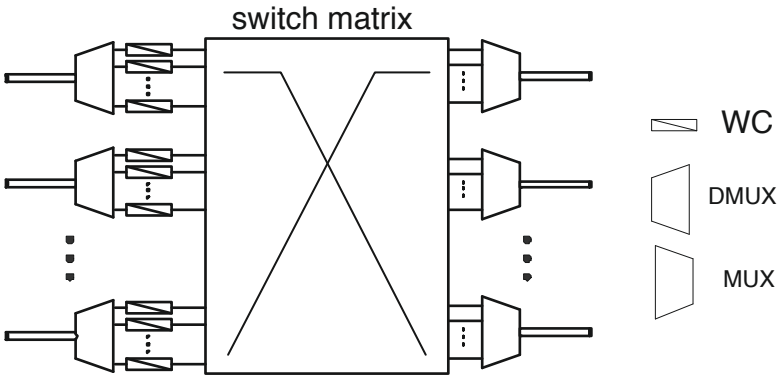

Figure 3.10: An OBS node with full wavelength conversion at input.

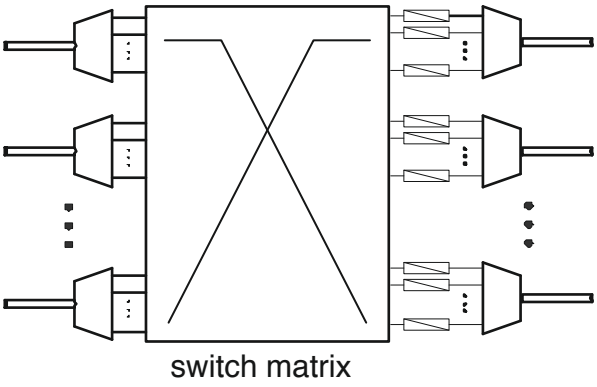

Figure 3.11: An OBS node with full wavelength conversion at output.

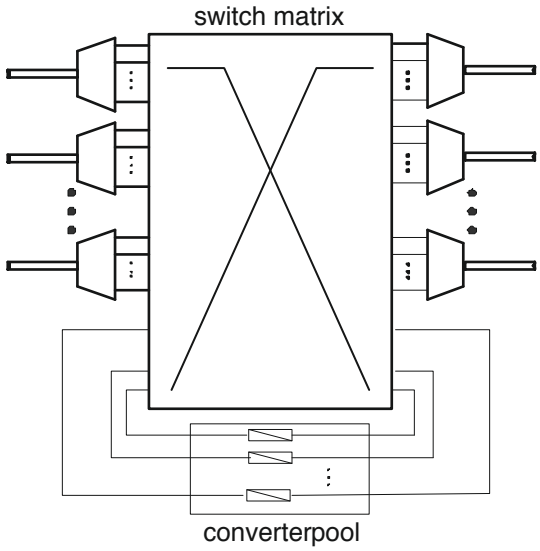

Figure 3.12: An OBS node with SPN partial wavelength conversion.

to achieve a certain blocking performance. The gain is particularly significant for the SPN case, where the converter pool is shared among all input channels of the node, see [112, 109].

Although wavelength conversion can to a high extent control the burst contention problem, the performance of an OBS node with a reasonable number of wavelength channels and converters is still far from being considered as acceptable [113]. Accordingly, this technique is usually used in tandem with other means of contention resolution schemes, such as FDL buffering.

3.4.2 FDL Buffering

In conventional packet-switched networks, buffering is the main approach for circumventing the contention problem. Applying the same technique to the optical domain is however a quite challenging task, since there is no equivalent to random access memory in the optical domain. Instead, optical delay loops can be used to delay packets in the times of contention. A delay loop is realized through a simple piece of fiber optic and therefore the provided delay is fixed and inflexible, in the sense that once a packet is forwarded to the delay loop it cannot be retrieved before it leaves the loop from the other end. The amount of delay provided in this way depends directly on the physical length of the loop, i.e., 5 μsec delay can be achieved per one kilometer of fiber[1].

A fiber delay line (FDL) buffer can be built by employing several basic delay lines of different lengths in a parallel structure [114, 115, 116, 117, 118]. Depending on the

[1]Light travels at the speed of 2×10^8 m/s inside the fiber optic.

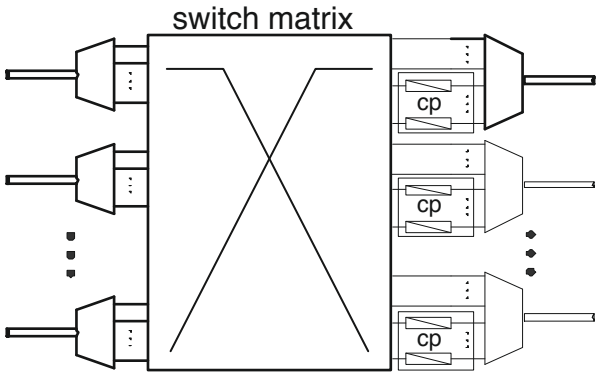

Figure 3.13: An OBS node with SPL partial wavelength conversion.

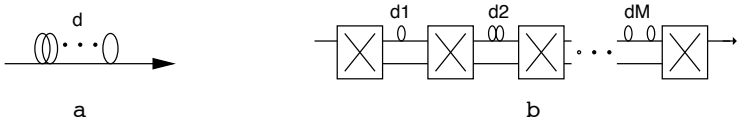

Figure 3.14: Basic element of an FDL buffer of type a) fixed delay, b) variable delay.

type of the basic delay lines applied, FDL buffers can be classified into two major categories, namely fixed-delay and variable-delay buffers [119]. In a fixed-delay buffer [120, 121], each basic delay line provides only a fixed delay as illustrated in Fig. 3.14-a. Alternatively, in a variable-delay buffer [94, 122], each basic delay line is a multistage structure, where each stage consists of a 2×2 switch and a fixed delay line as depicted in Fig. 3.14-b. In every stage the burst passes through either the associated delay line or the line of zero delay. This makes it possible to obtain different delay values by appropriately configuring states of the switches. Variable-delay FDL buffers can provide delay values with a much finer granularities compared to the fixed-delay ones, nonetheless, they are more complicated, and therefore more costly than fixed-delay buffers.

Depending on how to incorporate FDL buffers into a switch architecture, OBS nodes with FDL buffers can be categorized in two ways [119].

Feed-forward vs. Feed-back In a feed-forward architecture, buffering of a burst is performed while it traverses from input to output of the switch [114, 123, 124]. Accordingly, the bursts that are forwarded to the output of the switch will be either transmitted or dropped. However, in a feed-back architecture, bursts that are blocked by the desired output channel are fed back to the input of the switch through FDL buffers [114, 125, 126]. In this way a burst may be circulated between input and output of the switch several times until either it can be successfully transmitted over

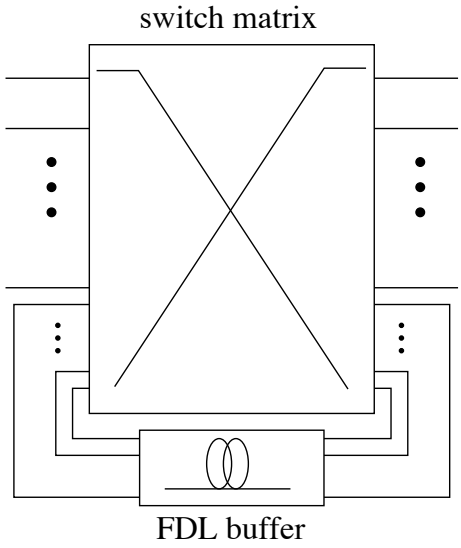

Figure 3.15: A switch with feed-back FDL buffer (single wavelength plane).

the desired channel or dropped. Fig. 3.15 shows a switch with FDL buffer utilized in the feed-back configuration.

Dedicated vs. Shared In an OBS switch with dedicated FDL architecture a separate buffering unit is dedicated to every output port of the switch. Alternatively, a single buffering unit can be shared among some or all output ports [127, 128, 129], as illustrated in Fig. 3.16 and Fig. 3.17.

Channel Scheduling At The Presence of FDL Buffers There are two general types of algorithms for scheduling bursts over the output channels of a switch at the presence of FDL buffers. Namely, there are FIFO and Non-FIFO algorithms [119]. In the former one the order of bursts remains unchanged, whereas in the latter one bursts reordering might frequently happen during buffering. To illustrate the difference between operation of the two algorithms, let us consider a fixed-delay buffer that is employed in the feed-forward architecture and discuss how bursts are delayed before transmission over the channel in case of contention.

Under the FIFO algorithm, when a new burst arrives to the system and there is at least one burst waiting in the buffer or being served over the channel, the scheduler calculates the earliest time t after which there would be no burst in the system. Then, the smallest idle delay line that provides a delay larger than t is assigned to the burst and the channel is accordingly reserved in advance. If there is no idle delay line that can provide the required delay, the burst has to be dropped or other contention resolution techniques have to be applied. By doing so, among all bursts in the buffer, the most

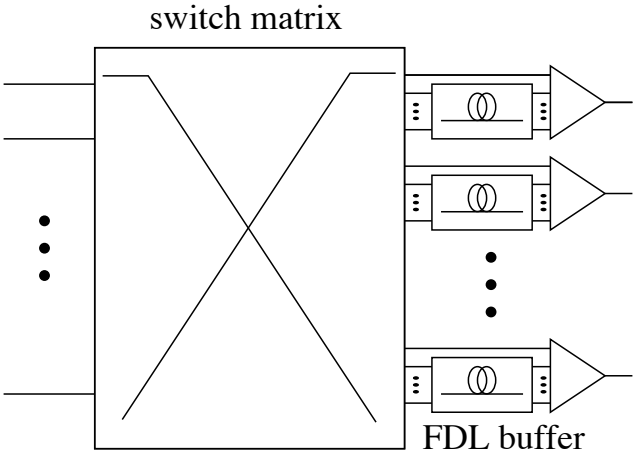

Figure 3.16: A switch with feed-forward dedicated FDL buffer (single wavelength plane).

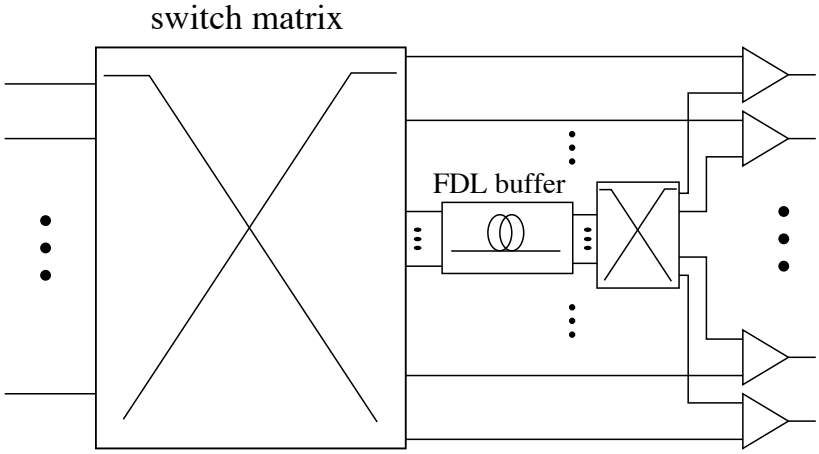

Figure 3.17: A switch with feed-forward shared FDL buffer (single wavelength plane).

recently enqueued one will experience the largest delay, thereby assuring that bursts will be served in the same order that they have arrived to the system.

By using the FIFO scheduling algorithm, bursts usually experience a delay that is larger than what is actually required to resolve a contention. As a result, some gaps are created between the time that the last bit of a burst leaves the channel and the time that next scheduled burst becomes ready for transmission. These gaps can reduce throughput of the system [130]. Accordingly, a Non-FIFO scheduling algorithm tends to utilize these gaps for transmission. That is, a new burst can be served in the gap between two already scheduled bursts. A Non-FIFO scheduling algorithm can achieve a better performance than a FIFO one. However, they might change the ordering of the bursts, that, in turn, can have negative impacts on performance of higher layer protocols like TCP [131]. Nevertheless, the issue of burst reordering and its influence on the TCP performance is still an open issue and needs further investigations. Furthermore, it can be shown that the smoothing effects of the burst assembly algorithms on the burst lengths can reduce the gain of using a Non-FIFO scheduling algorithm and make the system design more complicated.

3.4.3 Deflection Routing

In the last sections, wavelength conversion and FDL buffering were discussed as the main schemes that can resolve burst contentions in a core switching node. These two schemes rely on the additional resources employed in nodes. As an alternative approach to tackle the problem, one can resort to the network-wide contention resolution schemes. A major technique of this category is deflection routing, which relies on the use of whole links across the network as shared buffers. Specifically, in a core switching node that implements deflection routing, those bursts that lose the competition on accessing the appropriate channels (links) are intentionally forwarded over other wrong but idle links, in the hope that the downstream node will correct the direction of the burst so that the bursts will eventually get to their destination [11, 132, 133, 134, 135, 136, 137].

Deflection routing can be considered as a traffic engineering technique that also aims at balancing the traffic across the network. It is known that multi-path routing and load balancing techniques can be effective in distributing the traffic over all links of the network on the condition that source nodes have enough information about the loading situations at different parts of the network. Accordingly, to fully utilize the potential of deflection routing, each core node has to periodically receive information about the utilization of other links across the network. Otherwise, simply forwarding contending bursts to idle ports may in some situations even worsen the performance [132, 104, 138].

Another issue that needs particular attention is the interaction of deflection routing with offset-based signaling scheme. When an intermediate node decides to deflect the route of a burst, it has to check if this will increase the overall length of the path or

not. If this is the case, the offset time of the burst has to be appropriately increased to make sure that the burst will not overtake its corresponding BHP and that all downstream nodes will have enough time to process the BHP. This implies that some kind of buffering has to be used along with deflection routing in case the network is using offset-based signaling technique.

3.4.4 Hybrid Schemes

Although any of the three basic contention resolution schemes in optical networks, i.e., wavelength conversion, FDL buffering and deflection routing, can help reducing contentions and improving the performance, relying merely on one scheme might not suffice to achieve an acceptable performance or might be in some situations too costly. For instance, applying merely FDL buffers for contention resolution in an OBS network necessitates the use of a huge amount of bulky fibers, which is usually not feasible. Furthermore, equipping a switch with wavelength converters cannot suffice to overcome the contention problem if the number of wavelength channels is small, or on the other hand it can be too costly if the number of channels is large and the full wavelength conversion method is to applied. Also, it has been shown that there are situations that applying deflection routing cannot help, e.g., in a highly loaded network deflection routing does not improve the performance [104]. Consequently and to address the shortcomings of the basic contention resolution schemes, one can design a hybrid scheme by incorporating two or all the basic schemes [104, 49, 139]. This will add another degree of freedom in design and optimization of an OBS network. In fact the design of a hybrid contention resolution scheme is an optimization problem that can be formulated subject to two different objective functions. First, the objective function can be maximization of a certain performance metric, say burst drop rate or throughput, with a given set of available contention resolution resources. Alternatively, the optimization problem can be the minimization of the required contention resolution resources subject to a target performance metric.

It is possible to devise a hybrid scheme that incorporates all three basic contention resolution schemes, however, this can make the overall architecture of the OBS network too complicated and costly. Among all possible combinations there are two hybrid schemes that are considered more promising, namely, wavelength conversion with FDL buffering and wavelength conversion with deflection routing [119]. In other words, the wavelength conversion is used in both theses hybrid techniques. There are basically two reasons for this. First, the wavelength conversion scheme has a high potential for improving performance. Also, wavelength converters can easily be incorporated into the switch structure as compared for example to the FDL buffers. Between these two hybrid schemes, performance comparison studies have revealed that the option with FDL buffers outperforms the option with deflection routing [104].

Again, several hybrid contention resolution schemes can be devised by combining different types of FDL buffers and wavelength converters. For example, one can de-

sign a scheme incorporating a fixed-delay FDL buffer in the dedicated feed-forward architecture and a SPN wavelength converter pool. In addition, it is possible to use multiple stages of FDL buffers and wavelength converters. For instance, in a hybrid scheme with two stages of conversion and one stage of buffering, a burst arriving on wavelength w_a might undergo conversion in order to use FDL buffer on wavelength w_b, and then be transmitted over the outgoing link on wavelength w_c after being converted by the second stage of conversion. In a hybrid scheme with one stage of buffering and conversion, there are two possibilities for resolving a contention when it occurs. Namely, buffering may be applied before wavelength conversion or after it. In either case, when the contention is resolved in the first step, the second one is no longer considered. This concept can be utilized to optimize the operation of the system from different perspectives. For instance, by applying FDL buffering after conversion, one can design a contention resolution scheme that minimizes the delay experienced by bursts. Additionally, by appropriately selecting the number and the order of the stages one can design a hybrid scheme that minimizes the use of FDL buffer or wavelength converters for a given target drop rate, see [113, 140].

3.4.5 Other Schemes

The techniques introduced so far are the classical techniques to address collisions in optical networks. There are also other techniques that have been developed specifically to address the contention problem in the OBS network. Burst segmentation [141, 142] and load controlling/balancing [143, 144, 145, 146, 147].

In the burst segmentation technique [142], when a contention occurs between two data bursts, only the overlapping part of one of the bursts is dropped instead of dropping the burst completely. Specifically, each burst is composed of several segments, where a segment can contain one or more complete packets. Then, an intermediate node inside the network may drop one or more segments of a burst in order to resolve contention between two bursts. There are two options for dropping segments of bursts: head drop and tail drop. While in the former one segments at the head of the bursts may be dropped, in the latter one only segments at the tail of the bursts are subject to dropping. This option has been further used to develop QoS differentiation in the OBS network [141]. For instance, in order to give higher priority to some traffic flows in a network that implements the tail drop, packets belonging to the flows should be places at the head of the bursts. The issue associated with this approach is the implementation complexity that increases both at core nodes, e.g. segment boundary recognition and segmentation control, and at edge nodes, e.g. segmentation recovery.

In the load controlling/balancing techniques, the burst contentions in the network is reduced by limiting the load that is offered to the links or alternatively by distributing the load over the links. For instance, in [143] a closed-loop rate control technique is proposed, which is similar to TCP congestion control. Specifically, each node inside the network controls the intensity of load on each of its output links and report the

load values to the corresponding edge nodes. Edge nodes can then change the load that is injected into the network in such a way that the load offered to links inside the network does not exceed a given threshold.

Multipath routing is an approach to balancing the load in the network and thereby reducing the burst contentions. In multiphath routing, each source node maintains a list of possible paths to each destination and uses the least-loaded path in order to reduce the burst contention inside the network. The path changing can take place at burst level [144, 145, 146] or at flow level [147].

3.5 Objective of Thesis

Burst contention as outlined thus far is among the biggest challenges of the OBS network. In this dissertation we intend to address this issue. Our work is motivated by the following two observations:

- Most of the existing literature on this topic are centered on the reactive principle, i.e. they come into play only after the control unit of an OBS node, see Fig. 3.2, detects a conflict. Therefore, in this approach usually no effort is made to avoid the occurrence of burst contentions. To understand the reason behind this trend we recall that packet-based optical switching networks, in general, and OBS, in particular, are not native approaches for optics. Actually, the basic idea of optical packet/burst switching is borrowed from the electronic packet switching, which has proven successful, and a lot of effort has been put into tailoring packet switching to optics. However, some technological barriers do not allow one to one mapping of packet switching functionalities to optics, e.g. lack of optical buffers. In facing problems caused by such barriers the approaches that have usually been taken are in line with the electronic packet switching paradigm. In the specific case of packet contention, this has led to a huge amount of reactive contention resolution methods, since it is the approach that is taken in electronic networks. Nonetheless, we believe that a proactive contention control mechanism that is based on opportunities and limitations of the underlying optical network can complement the work in this field.

- Most of the works carried out so far in the direction of burst contention control look at the problem from a narrow perspective and consider this problem as a standalone one. For instance, in contention resolution mechanisms based on FDL buffers, the role of traffic characteristics, as shaped by the burst assembly unit, is usually overlooked and a great emphasis is put on the FDL buffer architecture. Nevertheless, we believe that broadening the scope of search – for contention control mechanism – and delegating an active role to the traffic in this game could help achieve this goal.

These two observations motivate us to consider the burst contention problem from a new perspective. Specifically, the focus of investigations in this work is put on development of a proactive contention avoidance/minimization mechanism based on traffic shaping in OBS network. For this purpose we address the following research questions:

- What is the traffic shaping effect of burst assembly process and how does it influence burst contention? To address this question we start with a thorough parameterization of traffic at the output of a single burst assembly queue, see Fig. 3.3, that implements a hybrid scheme. The main statistical characteristics of the traffic, e.g. distribution of burst size and burst interdeparture time are parameterized based on the size and time thresholds, i.e. L_{Th} and T_{Th}. Then, the results of this analysis is used in sets of simulation experiments to investigate the impacts of traffic shaping, through burst assembly, on the burst contention inside the network.
- How can the traffic shaping be leveraged for burst contention avoidance/minimization in the network? To address this question we utilize the insights that are gained while investigating the first research question and present an advanced framework for contention avoidance/minimization that is centered on traffic shaping. In our approach, which is called VOB, the traffic shaping takes place at the edge of the network, where we still have the chance to use inexpensive electrical buffers.

3.5.1 Assumptions

We consider an OBS network that operates under asynchronous mode and the structure of the nodes in the network is the same as the one depicted in Fig. 3.2. Additionally, we make the following assumptions concerning the network operation:

Burst Assembly: We consider that burst assembly units of the nodes in the network implement the hybrid burst assembly algorithm and that the size of generated bursts in the network are in the range of 10 KB. This assumption on the average burst size will allow the employment of FDL buffering as a reactive contention resolution method in the switches.

Signaling and Switching: The out-of-band signaling and store-and-forward switching method is used for this purpose. Also, an ideal synchronization is assumed between BHP packets and corresponding data bursts. Additionally, no switching or processing overhead is assumed for the nodes in the network.

Channel Reservation: For channel reservation we consider the non-void-filling version of JET protocol.

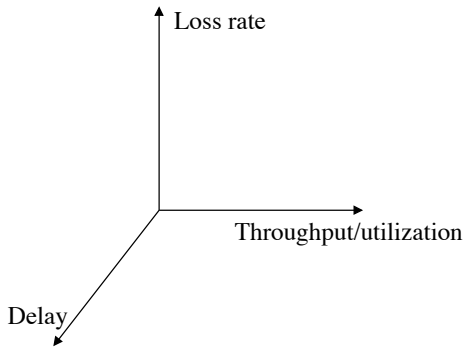

Figure 3.18: Performance evaluation space.

Contention Control: It is assumed that each node in the network is equipped with full wavelength conversion capability. Also, at the output ports of switches small FDL buffering is assumed.

Physical Layer Impairment: The physical layer of the network is assumed to be ideal, i.e. physical impairments of the signal are not considered.

3.5.2 Performance Metrics

To demonstrate the effectiveness of any proposal for contention control it is of great importance to analyze its operation and quantify its performance based on a clearly defined set of metrics. Figs. 3.18 shows the performance evaluation space considered in this work. According to the figure, we consider three metrics, namely, burst loss/drop rate, network-wide utilization and throughput as well as delay.

3.5.3 Burst Loss Rate

Loss rate – or drop rate – is the most important performance metric in evaluating a contention control technique and for a node/link is defined as the ratio of number of data units that are dropped at a core node/link due to lack of enough resources to the total number of data units received by that node/link. Therefore, loss rate could very well serve as a measure of effectiveness of a certain contention resolution technique in comparison to others. Depending on the definition of data units, loss rate could be considered at three levels: bit, packet or burst. While to evaluate the loss rate at bit level, we count the number of bits that are lost in the system, for packet and burst level we count the number of packet and bursts, respectively, that are dropped. In this work, burst loss rate is considered, and when it is meaningfully different from bit loss rates – because of different burst size – we consider bit loss rates.

3.5.4 Network Utilization and Throughput

Utilization and throughput at the network level are other metrics that should be looked at in analyzing any contention control scheme. Throughput of a network is the number of data units that are successfully received by their destination nodes in network in a unit of time – usually expressed as bursts or bits per second. Also, the utilization of the network is equal to the throughput of the network normalized to the network capacity. As it is in many cases difficult to determine the exact transport capacity of a specific network, the utilization is presented for different links of the network separately. The utilization of a link in the network is calculated as the ratio of the number of bits that are successfully transmitted over the link to the total transmission capacity of that link for a given time period in bits.

Utilization of a link is dependent, among others, on the drop rate P_D. Specifically, the utilization of a link is given by $U = \rho(1 - P_D)$, where ρ is the total load offered to that link as normalized to the transmission capacity of the link. The load offered to the link ρ is calculated as the total traffic received at input of the link as normalized to the total capacity of the link.

Loss rate of the system can easily be improved by decreasing the offered load to the system, i.e., under-utilizing the system. Therefore, it is crucial to assess the performance of the system at a reasonable utilization, say 50-70 % .

3.5.5 Delay

Delay is another relevant performance metric considered in this study. Delay T refers to the difference between the time a packet arrives at an OBS ingress node and the time when it leaves the network from an egress node. Apart from the propagation delay, which is usually the largest part of T in geographically large networks, the delay is composed of three parts. First, packets are delayed in the assembly buffer at ingress nodes. This delay is referred to as the assembly delay T_a and its value depends on the assembly mechanism applied in the ingress nodes. The second part of delay is associated with the queuing/shaping delay at the electrical burst queues at the edge of the network, T_q. The amount of T_q, among others, depend on the amount of transit traffic switched by the ingress OBS node. Finally, optical buffering at core nodes contributes to the delay a packet may experience in the network.

Part II

Analysis of Burst Assembly Process

Chapter 4

Burst Assembly Process: An Analytical Investigation

Assembly algorithms shape the statistical characteristics of the input traffic that, in turn, influence the performance of the network. In order to thoroughly analyze performance implications of the assembly process in OBS networks, the first step is to study the burst assembly algorithm itself. In this chapter we focus on analytical modeling of the burst assembly process as one of the main building blocks of OBS architecture. The work presented in this chapter consists of three parts:

- Analytical models are developed for length and interdeparture time distributions of the bursts that are generated by the most popular assembly algorithms. Namely, time-based, volume-based as well as hybrid burst assembly algorithms are considered. The analytical models are exact and among others include the probability that in the time-based and hybrid assembly, padding has to be applied in order that the burst length meets a given minimum length requirement, and the probability that in the hybrid assembler the timer expires.

- The models derived in the first part are synthesized and the technique known as *composition technique* is applied to them in order to develop simple algorithms that mimic real assembler's behavior by generating burst traffic of the same statistical characteristics. The algorithms can be exploited as a burst traffic generator in a per flow basis in discrete event simulation models to accelerate the simulation.

- The smoothing effect of the burst assembly algorithms on the traffic and its implications on the performance of the OBS operation are studied. Specifically, it is investigated how shaping the traffic at the edge of the OBS network – through varying the burst assembly parameters – can influence the burst contentions inside the network.

4.1 Related Work

Several studies have dealt with analytical modeling of burst traffic inside OBS networks [148, 149, 150, 151]. Unfortunately, these studies do not present a complete analysis of the most important burst assembly algorithm that simultaneously applies both time and volume thresholds, i.e. hybrid assembly algorithm.

The probability generating function (*pgf*) of burst length is derived in [148], under the assumption that the arrival process is the time-slotted Bernoulli. Then based on the pgf, the distribution of burst length is approximated by standard distributions such as the Gamma distribution. The analysis, however, does not include the distribution of burst interdeparture time. In [150], analytical models are given for distributions of both burst length as well as burst interdeparture time for different burst assembly strategies. Nevertheless, the work does not consider the hybrid assembly algorithm. The hybrid algorithm is studied in [149], where it is claimed that the burst length distribution would have a very narrow range, therefore only the distribution of burst interdeparture time is developed under the assumption that the burst length is fixed. In other words, the hybrid algorithm is approximated by the volume-based algorithm. In [151], a variation of the time-based assembly algorithm is studied, in which at the end of each aggregation period a length threshold is applied and contents of the buffer are accordingly assembled into a few bursts of a given length. The analysis provides the distribution of number of bursts that are generated at the end of an assembly period.

4.2 Burst Traffic Modeling

4.2.1 Assumptions

In developing analytical models, we make the common assumption that packet-level traffic arrives at the assembly buffer according to the Poisson process and packet lengths are exponentially distributed. These assumptions simplify our analysis and allow us to derive exact analytical models that can easily be used to synthesize burst traffic generators. In the next chapter, we conduct a trace-based analysis of the burst assembly in order to validate the major conclusions arrived at in this chapter. Furthermore, there are some recent measurements of traffic in the Internet suggesting that at sub-second time scales, packet arrivals in the core network follow the Poisson process [152]. We note here that the time scale associated with the burst assembly process is indeed far below a second.

Consider a single assembly buffer in an OBS ingress node, see Fig. 3.3. Let packets arrive to the buffer according to the Poisson process with rate λ, and also packet lengths be independent and identically distributed (*i.i.d.*) according to the exponential distribution with mean μ^{-1}, i.e., the offered load to the assembler is $\rho = \lambda\mu^{-1}$. In the following analysis we let X and Z be random variables denoting the length and interdeparture time of the bursts leaving the assembler's output, respectively.

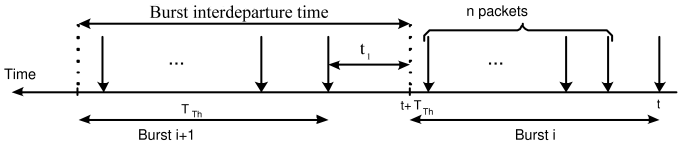

Figure 4.1: Burst formation process under the time-based assembly algorithm.

4.2.2 Time-Based Algorithm

Consider a time-based assembly algorithm with the assembly parameter and the minimum burst length requirement set to T_{Th} and L_{min}, respectively. According to Fig. 4.1, from the instant of time that the first packet arrives to the buffer (i.e. t), it takes T_{Th} units of time until a new burst will be generated. Therefore, we can assume that each burst contains $(N+1)$ packets where N is the total number of packets that arrive during the interval, after the first packet has arrived and before the timer expires. As a result, N would be a random integer that follows the Poisson distribution with mean λT_{Th}. That is,

$$P_T[N=n] = \frac{(\lambda T_{Th})^n}{n!} e^{-\lambda T_{Th}}. \tag{4.1}$$

The average number of packets per burst is equal to $\lambda T_{Th}+1$. If total number of packets in a burst is given by k, then length of the burst is sum of k *i.i.d.* exponentially distributed random variables that is known to follow the Erlang distribution [105] with the density function given by

$$f_T(x|k) = \frac{\mu(\mu x)^{k-1}}{(k-1)!} e^{-\mu x}. \tag{4.2}$$

Accordingly, the total volume of the packets in the buffer when the timer expires would have the following density function.

$$f_T(x) = \sum_{n=0}^{\infty} \frac{\mu(\mu x)^n}{n!} e^{-\mu x}.P_T[N=n] \quad (x>0). \tag{4.3}$$

The probability that the burst length is smaller than L_{min}, i.e., padding is required, would be equal to:

$$\begin{aligned} P_{T,Pad} &= P(x < L_{min}) \\ &= \int_0^{L_{min}} f_T(x)dx \\ &= \sum_{n=0}^{\infty} \frac{\gamma(n+1, \mu L_{min})}{n!} P_T[N=n] \end{aligned} \tag{4.4}$$

where $\gamma(a,x) = \int_0^x t^{a-1}e^{-t}dt$ is the incomplete gamma function. Therefore, the density function of bursts after padding would be given by

$$f_{T,Pad}(x) = P_{T,Pd}\delta(x - L_{min}) + \sum_{n=0}^{\infty} \frac{\mu(\mu x)^n}{n!} e^{-\mu x}.P_T[N = n] \quad (x \geq L_{min}) \tag{4.5}$$

where $\delta(.)$ is the Dirac delta function.

Now, we turn to the burst interdeparture time. As depicted in Fig. 4.1, by making use of the memoryless property of the packet arrival process, we notice that the interdeparture time between two consecutive bursts is composed of a random value t_1 that is exponentially distributed with mean λ^{-1} plus a deterministic value T_{Th}. The exponential part accounts for the time required until a new packet arrives to the buffer after the previous burst has been sent. In fact, the burst interdeparture time distribution is the same as the distribution of packet interarrival time, shifted by the assembly parameter. Therefore, the density function of burst interdeparture time can be written as the shifted-exponential density given in (4.6).

$$g_T(z) = \lambda e^{-\lambda(z - T_{Th})} \quad (z \geq T_{Th}). \tag{4.6}$$

Note that $f_T(z)$ is independent of μ^{-1}. In addition, while the random part would have a negligible effect on interdeparture times in a heavily loaded assembler, it plays an important role under the light load situations. That is, under small arrival rates, the random part may be comparable to the fixed part T_{Th}.

4.2.3 Volume-Based Algorithm

Let us now consider traffic generated by a volume-based assembly algorithm with assembly parameter L_{Th}. It is assumed that packet-level traffic with the same characteristics as described in the previous section arrives to the assembly buffer. Let us first derive the distribution function of burst interdeparture times.

Since both packet lengths and interarrival times are assumed to be exponentially distributed, the problem of finding distribution of burst interdeparture times can be regarded as equivalent to that of finding the burst length distribution for the time-based assembler with the assembly parameter L_{Th}, in which the packet arrival process is Poisson with mean μL_{Th} and packet lengths are exponentially distributed with mean λ^{-1}. Therefore, the density function of burst interdeparture times is given by

$$g_V(z) = \sum_{n=0}^{\infty} \frac{\lambda(\lambda z)^n}{n!} e^{-\lambda z}.P_L[N = n] \quad (z > 0). \tag{4.7}$$

where $P_L[N = n]$ is given by

$$P_V[N = n] = \frac{(\mu L_{Th})^n}{n!} e^{-\mu L_{Th}}. \tag{4.8}$$

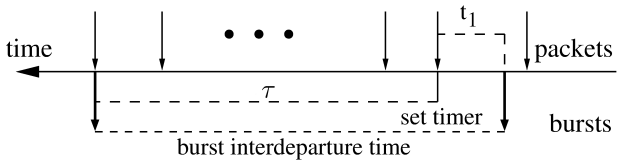

Figure 4.2: Burst formation process under the hybrid assembly algorithm.

Similar to the burst interdeparture times under the time-based assembler, burst length distribution in this case is the same as the distribution of packet length, shifted by the assembly parameter L_{Th}. Thus, the density function of burst length can be written as

$$f_V(x) = \mu e^{-\mu(x-L_{Th})} \ \ (x \geq L_{Th}). \tag{4.9}$$

In this case, $f_V(x)$ is independent of the packet arrival rate. Note that in viewing the process from this perspective, N, which has the Poisson distribution, is the total number of arrivals during an interval of length L_{Th}. However, as described in Chapter 3 a new burst will be generated upon arrival of the packet that makes the aggregate length exceed L_{Th}. Thus, the burst interdeparture time should be calculated using an Erlang distribution with $(n+1)$ phases. This also accounts for the exponential term in calculating the burst length. As a result, the average number of packets per burst in this case is equal to $\mu L_{Th} + 1$.

4.2.4 Hybrid Algorithm

Consider a hybrid algorithm in which the minimum burst length requirement, volume threshold and time threshold are set to L_{min}, L_{Th} and T_{Th}, respectively, where $L_{min} < L_{Th}$. As depicted in Fig. 4.2, once a packet enters an empty assembly buffer, the timer is set to T_{Th}. Therefore, the probability that the timer expires before the volume criterion L_{Th} is met can be expressed as

$$P_{Tout} = 1 - P(\tau < T_{Th}) \tag{4.10}$$

where τ is the random variable characterizing the time needed until enough packets arrive to the assembly buffer, so that a burst can be generated using the volume criterion. We recall from the volume-based assembler that the conditional pdf of τ given the number of arrivals follows the Erlang distribution. That is,

$$h(\tau \,|\, n) = \begin{cases} \delta(\tau), & n = 0 \\ \frac{\lambda(\lambda\tau)^{n-1}}{(n-1)!} e^{-\lambda\tau}, & n \geq 1 \end{cases} \tag{4.11}$$

where n is the number of arriving packets whose aggregate length is L_{Th}. The probability function of n is given in (4.8). Note that $n = 0$ is associated with the situation

that the size of the first arrival to the buffer is larger than the volume threshold. In that case, the timer will be deactivated immediately after it is set, thus τ is equal to zero. For the case of $n \geq 1$ we would have

$$\begin{aligned} P(\tau < T_{Th} \,|\, n) &= \int_0^{T_{Th}} \frac{\lambda(\lambda\tau)^{(n-1)}}{(n-1)!} e^{-\lambda\tau} d\tau \\ &= \frac{\gamma(n, \lambda T_{Th})}{(n-1)!} \quad (n \geq 1). \end{aligned} \tag{4.12}$$

Therefore, P_{Tout} can be computed as

$$P_{Tout} = 1 - e^{-\mu L_{Th}} - \sum_{n=1}^{\infty} \frac{\gamma(n, \lambda T_{Th})}{(n-1)!} P_V[N = n]. \tag{4.13}$$

Now, let us derive the pdf of length of the bursts that are generated by the assembly buffer under study. The bursts that leave the assembly buffer before the timer expires would be larger than L_{Th} in length, thus the pdf of their length would be equivalent to that of the bursts generated by the pure volume-based algorithm, as given in (4.9). On the other hand, if the timer does expire, the burst length would be smaller than L_{Th}. Therefore, the pdf of burst length before possible padding would be

$$f_H(x) = \begin{cases} P_{Tout} f_a(x), & x < L_{Th} \\ (1 - P_{Tout}) \mu e^{-\mu(x - L_{Th})}, & x \geq L_{Th} \end{cases} \tag{4.14}$$

where $f_a(x)$ is the pdf of the burst length when the timer expires, which is equal to the conditional pdf of $f_T(x)$, as given by (4.3) , given $(x < L_{Th})$. That is,

$$\begin{aligned} f_a(x) &= f_T(x \,|\, x < L_{Th}) \\ &= \frac{f_T(x)}{K_1} \quad (x < L_{Th}) \\ &= \frac{1}{K_1} (\sum_{n=0}^{\infty} \frac{\mu(\mu x)^n}{n!} e^{-\mu x} P_T[N = n]) \quad (x < L_{Th}) \end{aligned} \tag{4.15}$$

where K_1 is the normalization factor and can be computed as, (see (4.4))

$$\begin{aligned} K_1 &= P(X < L_{Th}) \\ &= \sum_{n=0}^{\infty} \frac{\gamma(n+1, \mu L_{Th})}{n!} P_T[N = n]. \end{aligned} \tag{4.16}$$

To calculate the probability that padding is applied we can write,

$$\begin{aligned} P_{H,Pad} &= P(X < L_{min}) \\ &= \int_0^{L_{min}} f_H(x) dx \quad (L_{min} < L_{Th}) \end{aligned}$$

$$= \int_0^{L_{min}} P_{Tout} f_a(x) dx$$
$$= \frac{P_{Tout}}{K_1} (\sum_{n=0}^{\infty} \frac{\gamma(n+1, \mu L_{min})}{n!} P_T[N=n]). \quad (4.17)$$

Finally, the pdf of burst length after padding can be expressed as

$$f_{H,Pad}(x) = \begin{cases} P_{H,Pad}\delta(x - L_{min}) + P_{Tout} f_a(x), & L_{min} \le x < L_{Th} \\ (1 - P_{Tout})\mu e^{-\mu(x-L_{Th})}, & x \ge L_{Th}. \end{cases} \quad (4.18)$$

Now we turn to the burst interdeparture time. As depicted in Fig. 4.2 burst interdeparture time is composed of two parts, namely t_1, which is the time required until a packet arrives after the last burst has been sent out, and τ, which is the time between when the first packet arrives and the new burst is generated. That is,

$$Z = t_1 + \tau. \quad (4.19)$$

For those bursts that are released due to the timer expiration, τ would be deterministic and equal to T_{Th}, thus the pdf of burst interdeparture time would be equal to that of the pure time-based assembler, as given in (4.6). For other bursts, however, interdeparture time would be equal to sum of t_1 and τ given $\tau < T_{Th}$. Therefore,

$$g_H(z) = P_{Tout}\lambda e^{-\lambda(z-T_{Th})} U(z - T_{Th}) + (1 - P_{Tout}) g_a(z) \quad (z > 0) \quad (4.20)$$

where $U(.)$ is the unit step function. Also, $g_a(z)$ can be calculated by convolving the density functions of t_1 and τ as follows:

$$g_a(z) = s(t_1) * h(\tau \,|\, \tau < T_{Th}) \quad (4.21)$$

where due to the Poisson packet arrival, the pdf of t_1 is given by $s(t_1) = \lambda e^{-\lambda t_1}$ $(t_1 \ge 0)$. Also, from (4.11) and (4.12) we would have

$$h(\tau) = e^{-\mu L_{Th}} \delta(\tau) + \sum_{n=1}^{\infty} \frac{\lambda(\lambda\tau)^{n-1}}{(n-1)!} e^{-\lambda\tau} P_V[N=n]. \quad (4.22)$$

Therefore, we can write,

$$h(\tau \,|\, \tau < T_{Th}) = \frac{h(\tau)}{P(\tau < T_{Th})} \quad (\tau < T_{Th}) \quad (4.23)$$

where

$$P(\tau < T_{Th}) = e^{-\mu L_{Th}} + \sum_{n=1}^{\infty} \frac{\gamma(n, \lambda T_{Th})}{(n-1)!} P_V[N=n]. \quad (4.24)$$

Substituting (4.23) in (4.21) and solving the convolution yields

$$\begin{aligned} g_a(z) &= s(t_1) * h(\tau \,|\, \tau < T_{Th}) \\ &= \int_0^\infty s(z-\theta) h(\theta \,|\, \theta < T_{Th}) d\theta \\ &= \frac{1}{P(\tau < T_{Th})} \int_0^\infty \lambda e^{-\lambda(z-\theta)} (e^{-\mu L_{Th}} \delta(\theta) \\ &+ \sum_{n=1}^{\infty} \frac{\lambda(\lambda\theta)^{n-1}}{(n-1)!} e^{-\lambda\theta} P_V[N=n]) d\theta \end{aligned} \tag{4.25}$$

In case $z \leq T_{Th}$, we would have

$$g_a(z) = \frac{1}{P(\tau < T_{Th})} (\lambda e^{-\mu L_{Th}} e^{-\lambda z} + \sum_{n=1}^{\infty} \frac{\lambda(\lambda z)^n}{n!} e^{-\lambda z} P_V[N=n]) \quad (0 < z \leq T_{Th}). \tag{4.26}$$

Otherwise,

$$g_a(z) = \frac{1}{P(\tau < T_{Th})} (\lambda e^{-\mu L_{Th}} e^{-\lambda z} + \sum_{n=1}^{\infty} \frac{\lambda(\lambda T_{Th})^n}{n!} e^{-\lambda z} P_V[N=n]) \quad (z > T_{Th}). \tag{4.27}$$

Therefore,

$$g_a(z) = \begin{cases} \frac{1}{P(\tau<T_{Th})} (\lambda e^{-\mu L_{Th}} e^{-\lambda z} + \sum_{n=1}^{\infty} \frac{\lambda(\lambda z)^n}{n!} e^{-\lambda z} P_V[N=n]) & 0 < z \leq T_{Th} \\ \frac{1}{P(\tau<T_{Th})} (\lambda e^{-\mu L_{Th}} e^{-\lambda z} + \sum_{n=1}^{\infty} \frac{\lambda(\lambda T_{Th})^n}{n!} e^{-\lambda z} P_V[N=n]) & z > T_{Th}. \end{cases} \tag{4.28}$$

Accordingly, burst interdeparture time can be expressed as

$$g_H(z) = \begin{cases} P_{Tout} \lambda e^{-\lambda(z-T_{Th})} \\ \quad + \frac{1-P_{Tout}}{P(\tau<T_{Th})} (\lambda e^{-\mu L_{Th}} e^{-\lambda z} + \sum_{n=1}^{\infty} \frac{\lambda(\lambda T_{Th})^n}{n!} e^{-\lambda z} P_V[N=n]) & z > T_{Th} \\ \frac{1-P_{Tout}}{P(\tau<T_{Th})} (\lambda e^{-\mu L_{Th}} e^{-\lambda z} + \sum_{n=1}^{\infty} \frac{\lambda(\lambda z)^n}{n!} e^{-\lambda z} P_V[N=n]) & 0 < z \leq T_{Th}. \end{cases} \tag{4.29}$$

After having derived exact expressions for the pdf of burst interdeparture time, we now take some practical considerations into account and approximate P_{Tout} and $g_H(z)$ with two simpler expressions. In practice, the length threshold L_{Th} is usually much larger than μ^{-1}, so that $e^{-\mu L_{Th}} \simeq 0$. As a result, P_{Tout} can be approximated by

$$P_{Tout} \simeq 1 - \sum_{n=1}^{\infty} \frac{\gamma(n, \lambda T_{Th})}{(n-1)!} P_V[N=n]. \tag{4.30}$$

Simplifying (4.29) and substituting P_{Tout} with the expression in (4.30), $g_H(z)$ can be approximated by

$$g_H(z) \simeq \begin{cases} (1 - \sum_{n=1}^{\infty} \frac{\gamma(n,\lambda T_{Th})}{(n-1)!} P_V[N=n]) \lambda e^{-\lambda(z-T_{Th})} \\ \quad + \sum_{n=1}^{\infty} \frac{\lambda(\lambda T_{Th})^n}{n!} e^{-\lambda z} P_V[N=n] & z > T_{Th} \\ \sum_{n=1}^{\infty} \frac{\lambda(\lambda z)^n}{n!} e^{-\lambda z} P_V[N=n] & 0 < z \leq T_{Th}. \end{cases} \tag{4.31}$$

4.3 Generation of Burst Traffic in a Discrete Event Simulator

In this section, we analyze the density functions that were derived in the last section, in order to develop efficient algorithms that take care of the burst traffic generation process in a discrete event simulation model. Let us start with the time-based burst assembly. The density function in (4.5) is composed of two parts, where the second part includes the product of two functions namely, the Erlang density function with $(n+1)$ phases and mean $(n+1)\mu^{-1}$ and the Poisson distribution with mean λT_{Th}. Moreover, we notice that the second part in (4.5) constitutes a *convex* combination of an infinite number of Erlang density functions, each with different mean and number of phases. By definition, a density function $f(y)$ is said to be a convex combination of other density functions f_1, f_2... if it can be written as

$$f(y) = \sum_{i=0}^{\infty} f_i(y) p_i \tag{4.32}$$

where $p_i \geq 0$, $\sum_{i=0}^{\infty} p_i = 1$, and each f_i is a density function [153].

An interesting feature of a complex density function that can be expressed in the form of a convex combination of other simple density functions, like that in (4.5), is that its samples can be generated using the *composition* technique [153]. In this technique, each sample of a random variable Y with the density function $f(y)$ is generated by: *i)* generating a positive random integer I such that $P(I = i) = p_i$ where $i = 0, 1, ...,$ and *ii)* returning Y with the density function $f_I(y)$. More specifically, a sample of random variable with the density function given in (4.5) can be easily generated in two steps. First, a positive random integer n is generated using the Poisson distribution with mean λT_{Th}, then a random value is generated using the Erlang distribution with $(n+1)$ phases and mean $(n+1)\mu^{-1}$.

Algorithm 1 presents the pseudo code of a simple algorithm that exploits the concept of the composition technique to generate burst traffic with burst length and interdeparture time density functions that are given in (4.5) and (4.6), respectively. Referring to the algorithm, $exp(\alpha)$, $Poisson(\beta)$ and $Erlang(n, n\alpha)$ are an exponential random generator with mean α, a Poisson random generator with mean β and an Erlang random generator with n phases and mean $n\alpha$, respectively. The algorithm works as follows. First, the number of packets in the current burst is determined. Then, the length of the current burst is computed using the *Erlang* random generator and taking into account that the length should not be smaller than L_{min}. Finally, interdeparture time is computed by generating an exponentially distributed random variable and adding it to T_{Th}.

Following the same approach for the volume-based case, we will get the algorithm depicted in Algorithm 2, that works in a similar way as the time-based algorithm.

To develop an algorithm for generating bursts with the hybrid policy, which is more complicated than the time and volume based cases, we use a combination of the first

Algorithm 1 Burst traffic generation according to the time-based algorithm with packet arrival rate λ, average packet size μ^{-1} and assembly parameters T_{Th} and L_{min}.

$\lambda, \mu, T_{Th}, L_{min} \leftarrow$ initialize
$k \leftarrow$ number of bursts to be generated
for $i := 1, i \leq k, i++$ **do**
  $n \leftarrow Poisson(\lambda T_{Th})$
  $x \leftarrow Erlang(n+1, (n+1)\mu^{-1})$
  if $x < L_{min}$ **then**
    $x \leftarrow L_{min}$
  end if
  $z \leftarrow T_{Th} + exponential(\lambda^{-1})$
  wait (z)
  generate and send a burst of length x
end for

two algorithms as depicted in Algorithm 3. The algorithm works as follows. In the first step, the value of P_T has to be computed using (4.13) or (4.30). Note that the value of P_T for a given set of input traffic and assembly parameters is fixed, thus it is computed only once at the beginning of the simulation. After having computed t_1 (see Fig. 4.2), a sample is drawn from a uniform random generator in the range $(0, 1)$. The value of U will then be used to decide which procedure, between time-based and volume-based, has to be followed to generate the current burst. In the former case, care has to be taken so as not to generate a burst whose length is larger than L_{Th}. This is achieved through a conditional probability. Similarly, in the latter case the burst interdeparture time minus t_1 must be smaller than T_{Th}. In either case, the conditional probability is achieved through implementing a while loop.

Algorithm 2 Burst traffic generation according to the volume-based algorithm with packet arrival rate λ, average packet size μ^{-1} and assembly parameters L_{Th}.

$\lambda, \mu, L_{Th} \leftarrow$ initialize
$k \leftarrow$ number of bursts to be generated
for $i := 1, i \leq k, i++$ **do**
  $n \leftarrow Poisson(\mu L_{Th})$
  $z \leftarrow Erlang(n+1, (n+1)\lambda^{-1})$
  $x \leftarrow L_{Th} + exponential(\mu^{-1})$
  wait (z)
  generate and send a burst of length x
end for

The main advantage of using the presented algorithms over direct implementation of the burst assembly algorithms is simulation speed-up. Specifically, if the assembly algorithm is directly implemented in a simulation model, the model first has to generate

Algorithm 3 Burst traffic generation according to the hybrid algorithm with packet arrival rate λ, average packet size μ^{-1} and assembly parameters T_{Th}, L_{Th} and L_{min}.

$\lambda, \mu, T_{Th}, L_{Th}, L_{min} \leftarrow$ initialize
$k \leftarrow$ number of bursts to be generated
$P_T \leftarrow$ compute P_T from the model
for $i := 1, i \leq k, i++$ **do**
  $t_1 \leftarrow exponential(\lambda^{-1})$
  $U \leftarrow uniform(0, 1)$
  if $U \leq P_T$ {use time-based procedure} **then**
    $x \leftarrow L_{Th} + 1$
    while $x > L_{Th}$ **do**
      $n \leftarrow Poisson(\lambda T_{Th})$
      $x \leftarrow Erlang(n + 1, (n + 1)\mu^{-1})$
    end while
    if $x < L_{min}$ **then**
      $x \leftarrow L_{min}$
    end if
    $z \leftarrow T_{Th} + t_1$
  else {use volume-based procedure}
    $z \leftarrow T_{Th} + 1$
    while $z > T_{Th}$ **do**
      $n \leftarrow Poisson(\mu L_{Th})$
      $z \leftarrow Erlang(n, n\lambda^{-1})$
    end while
    $z \leftarrow z + t_1$
    $x \leftarrow L_{Th} + exponential(\mu^{-1})$
  end if
  wait (z)
  generate and send a burst of length x
end for

a large number of packets, and then assemble them into bursts. This process has to be repeated for every single burst. In other words, it would take $O(MN)$ time to generate N bursts, where M is the average number of packets per burst. This, however, implies that the more that the average number of packets per burst is, the slower the corresponding simulation model works. This is fortunately not the case for our algorithms, in which time needed to generate N bursts is $O(N)$. In the next section, we will numerically evaluate the speed-up gain that is achieved using these algorithms.

4.4 Numerical Analysis

In this section, characteristics of traffic generated by the burst assemblers are numerically studied and discussed. In order to validate the correctness of the analysis we have first simulated the system with the same assumptions made for the analysis. It was observed that the simulation results exactly match those obtained from the analysis as well as those achieved when the proposed burst generation algorithms are implemented. This was not surprising as the analytical models derived in this work are exact. As a result and for the sake of brevity, only the numerical results based on the developed models are presented here. For this purpose, we focus on the hybrid assembler as the other two types can be considered as special cases of the hybrid algorithm.

Let us consider a burst assembly queue that is fed by a traffic flow whose data rate is 25 Mb/s and the average size of input packets is equal to 645 B. These numbers are selected such that they are consistent with the trace-based case study presented in Chapter 5.

The hybrid assembler applies a combination of time and size criteria to decide for generating bursts, thus it is important to see the interplay between these two criteria for different settings. In Fig. 4.3, values of P_{Tout} are plotted as a function of T_{Th} for different values of length threshold L_{Th}. Observe that, as expected, the range of variation of T_{Th} when P_{Tout} varies from 0 to 1 increases with L_{Th}. In fact, at smaller values of L_{Th}, e.g. at $L_{Th} = 10$ KB a small change in T_{Th} greatly affects the fraction of bursts that are generated due to timer expiry, though this sensitivity to T_{Th} decreases with L_{Th}.

Now let us consider distributions of burst size and burst interdeparture time. For this purpose we consider several scenarios based on two values of L_{Th}= 10, 20 KB and different values of T_{Th}. The selection of T_{Th} values are done in a way that they lead to P_{Tout}=0, 0.25, 0.5, 075. The corresponding T_{Th} values are depicted in Table 4.1.

Figs. 4.4 and 4.5 depict the distribution of burst size at the output of the burst assembler for the considered scenarios. The distributions in Figs. 4.4 and 4.5 can be divided into two regions, with the border of the regions being at the length threshold of the assembler, i.e. 10 or 20 KB. The first region (burst length smaller than the length threshold) is associated with those bursts that are generated when the timer expires.

Table 4.1: Values of T_{Th} for the considered experiments (ms).

	$P_{Tout} = 0$	$P_{Tout} = 0.25$	$P_{Tout} = 0.5$	$P_{Tout} = 0.75$
$L_{Th} = 10$ KB	∞	3.912	3.096	2.374
$L_{Th} = 20$ KB	∞	7.436	6.297	5.252

The second region, however, is associated with bursts that are generated because the length threshold is exceeded. The ratio between the number of bursts in two regions is determined by P_{Tout}. Therefore, P_{Tout} is the key parameter in characterizing the burst length distribution. Note that P_{Tout} itself is a function of the input traffic rate and the selected time and length thresholds of the assembler. Table 4.2 depicts the mean and the variance values of the burst length for the considered scenarios. It is observed that for a given assembly threshold L_{Th} average burst length decreases and the variance increases with P_{Tout}. In fact, the coefficient of variation of burst size increases with P_{Tout} and also with L_{Th} for a given P_{Tout}. For instance, at $P_{Tout} = 0.5$ the CoV of burst length decreases from 0.227 to 0.157 when L_{Th} increases from 10 KB to 20 KB.

Corresponding burst interdeparture time distributions are shown in Figs. 4.6 and 4.7. Similar to the burst length, the distribution of burst interdeparture times can also be divided into two regions, with the border of the regions being at the time equal to the time threshold of the assembler. Also, Table 4.3 depicts the mean and the variance values of the burst interdeparture times. Observe that, in contrast to burst length, the CoV of the burst interdeparture times decreases with P_{Tout}. In addition, for a given P_{Tout}, the CoV decreases with L_{Th}, that is, increasing the length threshold improves the smoothness of the burst interdeparture times.

4.4.1 Smoothing Impact of Assembly

Now let us consider the smoothing effect of burst assembly process on the traffic. For this purpose, we use the algorithms developed in Section 4.3 and numerically estimate the coefficient of variation of the number of burst departures against the time scale for the considered traffic flow and at different values of P_{Tout}. The results are depicted in Figs. 4.8 and 4.9 for L_{Th}= 10 KB and 20 KB, respectively. Each plot also includes CoV curves of fictitious burst flows, whose burst arrivals follow the Poisson process and have the same burst arrival rates as the corresponding assembled traces.

It is observed that the CoV of the number of burst arrivals for the assembled traffic is significantly smaller than the corresponding Poisson process for the considered scenarios. Also, the CoVs reduce with P_{Tout}, which is due to the fact that CoV of burst interdeparture times reduces with P_{Tout}. In fact, there is a direct relation between the CoV of burst interdeparture times and that of number of burst departures at a given time scale.

Table 4.2: Calculated mean and variance of burst length for different scenarios.

	$P_{Tout} = 0$		$P_{Tout} = 0.25$		$P_{Tout} = 0.5$		$P_{Tout} = 0.75$	
L_{Th}	Mean(B)	Var(B^2)	Mean(B)	Var(B^2)	Mean(B)	Var(B^2)	Mean(B)	Var(B^2)
10 KB	10644	4.14×10^5	9997	2.11×10^6	9058	4.2×10^6	7642	6.02×10^6
20 KB	20644	4.14×10^5	19774	3.8×10^6	18459	8.4×10^6	16407	1.3×10^7

Table 4.3: Calculated mean and variance of burst interdeparture times for different scenarios.

	$P_{Tout} = 0$		$P_{Tout} = 0.25$		$P_{Tout} = 0.5$		$P_{Tout} = 0.75$	
L_{Th}	Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)
10 KB	3.409	1.36	3.2	0.725	2.9	0.356	2.45	0.136
20 KB	6.61	2.69	6.33	1.49	5.91	0.75	5.25	0.27

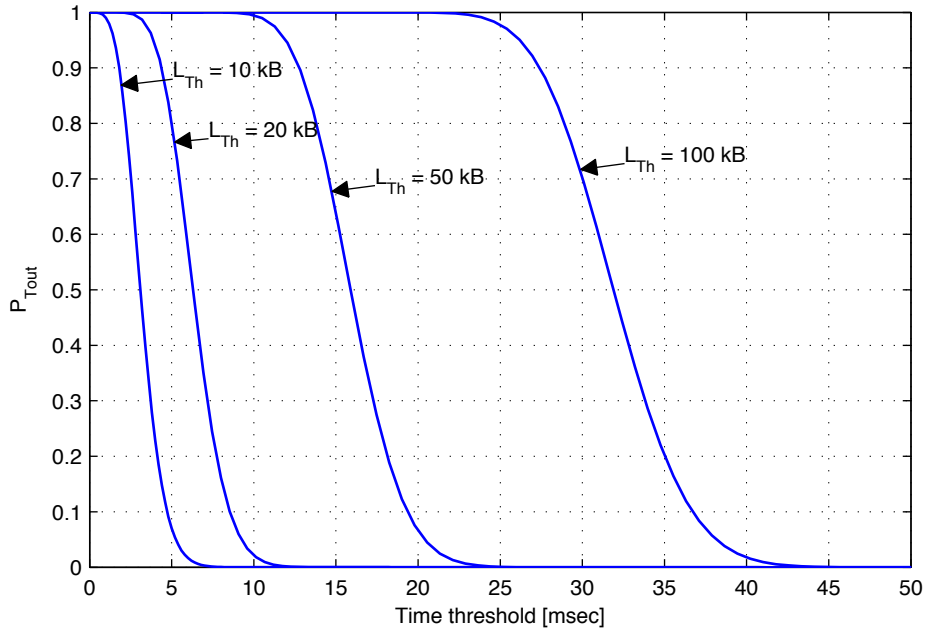

Figure 4.3: Probability of the assembly timer expiry as a function of time threshold T_{Th} at different values of L_{Th}.

4.4.2 Speed-up Gain of Traffic Generation Algorithms

After having discussed the characteristics of the burst traffic, we now study the speed-up gain that is achieved when using the algorithms of Section 4.3 in a simulation model. The proposed algorithms are based on exact analytical models that have been developed in Section 4.2, thus the traffic that is generated using such algorithms exactly match those of analytical models. Therefore, here we only present and discuss the speed-up gain of such algorithms over straightforward implementation of the assembly algorithms in a simulation model.

For this purpose, traces of burst traffic are generated through the direct approach as well as the proposed algorithms and the corresponding simulation times are compared with each other. Both algorithms are implemented in OMNeT++, in which all basic random number generators required for implementing the algorithms are available as built-in functions. We have measured and compared time needed for each of the approaches in order to generate 10^7 bursts on a PentiumIV 3.2GHZ processor with 1024 MB RAM. Fig. 4.10 presents the speed-up gain that has been measured versus the average number of packets per burst. It is seen that the speed-up gain for the pure time-based and volume-based algorithms could be well beyond an order of magnitude even for short bursts, and it increases with the average number of packets per burst. The reason is that in the direct implementation, as discussed before, simula-

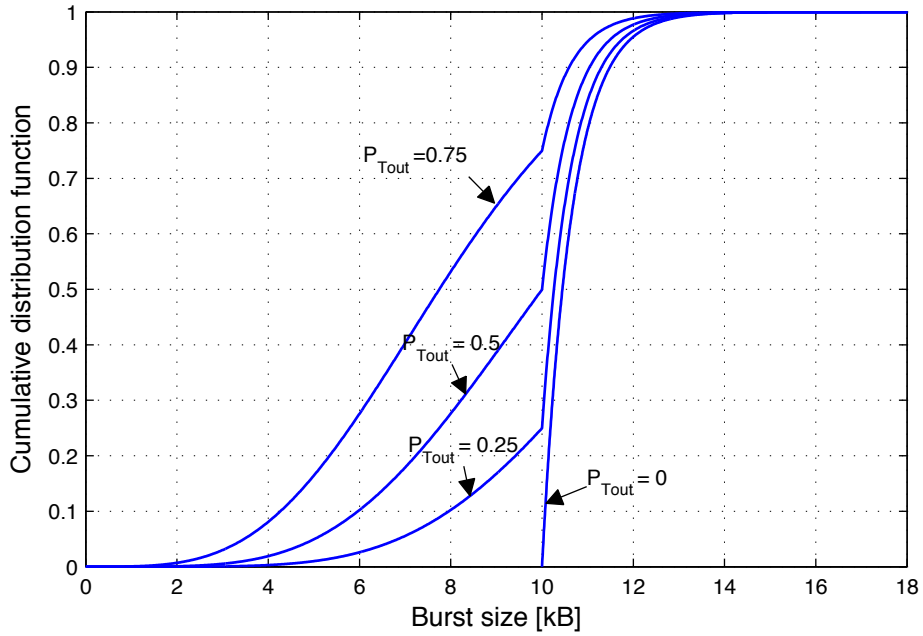

Figure 4.4: CDF of burst size at L_{Th}=10 KB.

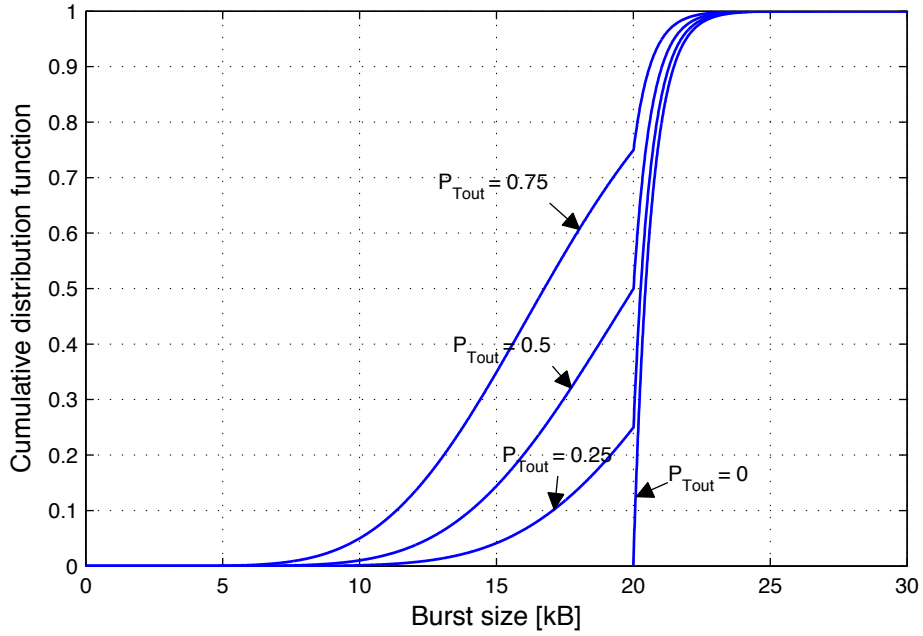

Figure 4.5: CDF of burst size at L_{Th}=20 KB.

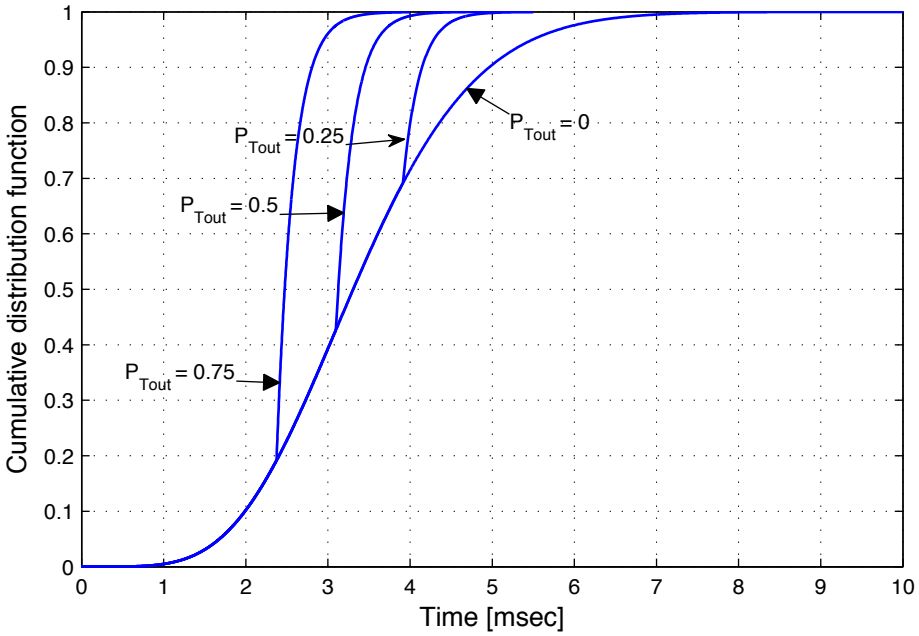

Figure 4.6: CDF of burst interdeparture times at L_{Th}=10 KB.

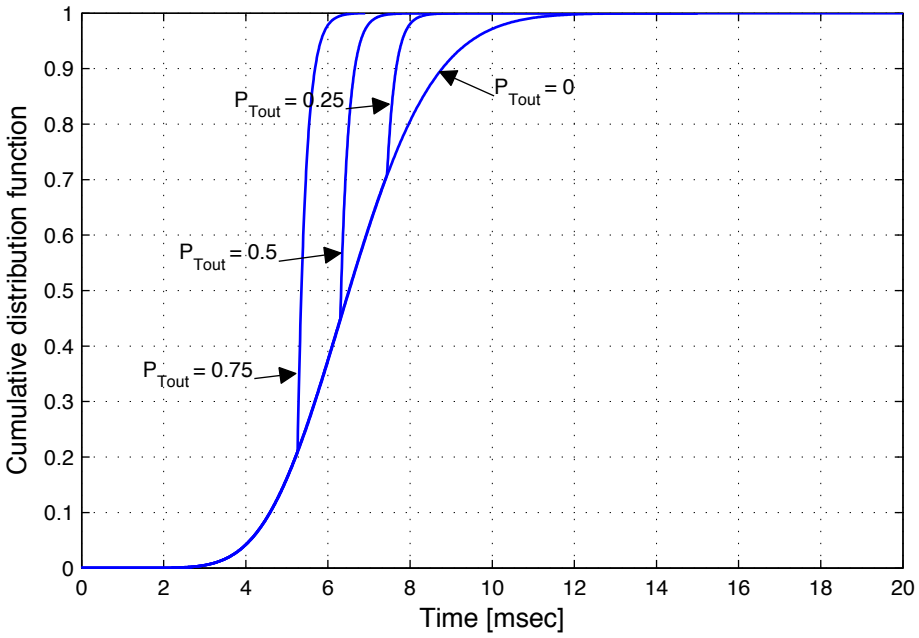

Figure 4.7: CDF of burst interdeparture times at L_{Th}=20 KB.

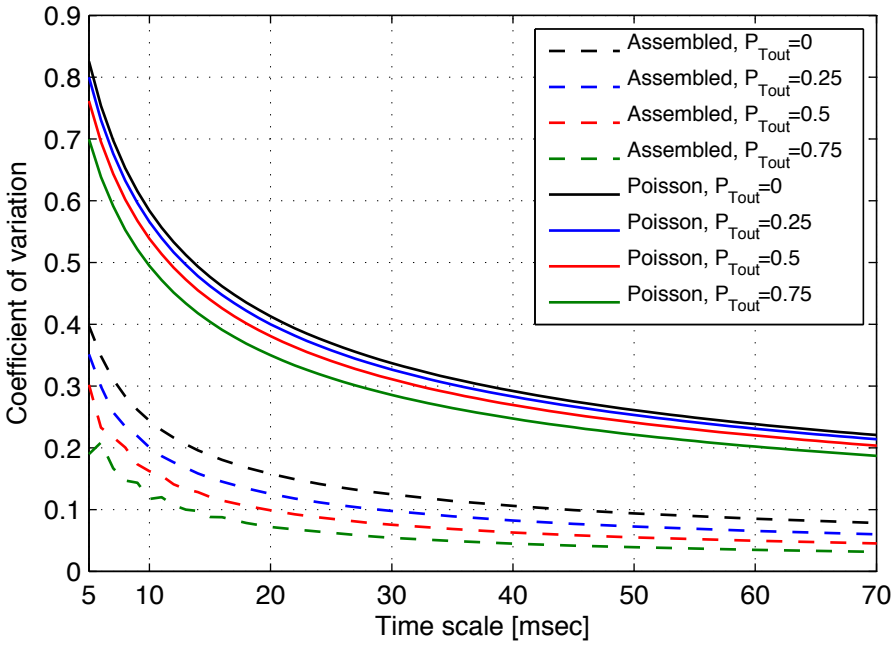

Figure 4.8: Coefficient of variation of number of burst departures as a function of time scale at L_{Th}=10 KB.

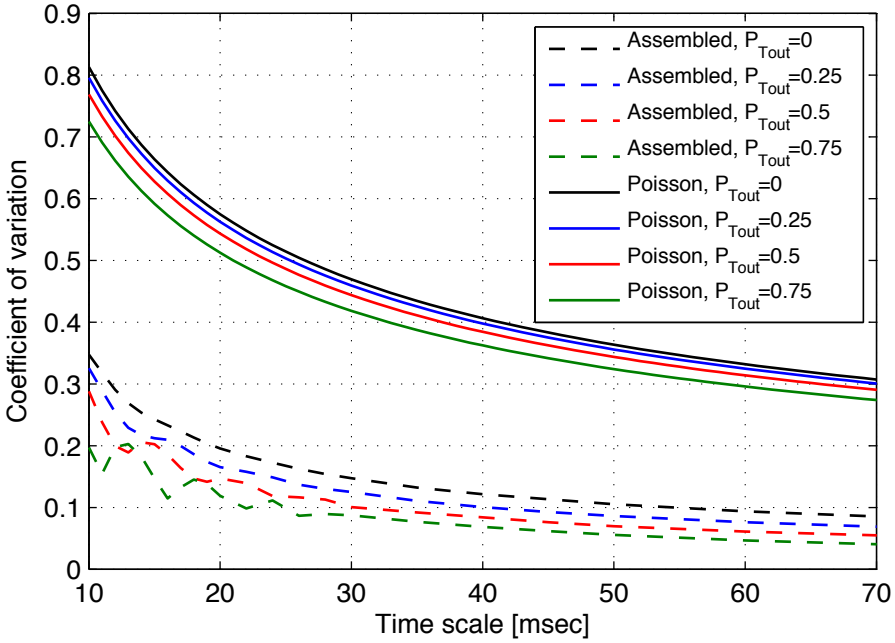

Figure 4.9: Coefficient of variation of number of burst departures as a function of time scale at L_{Th}=20 KB.

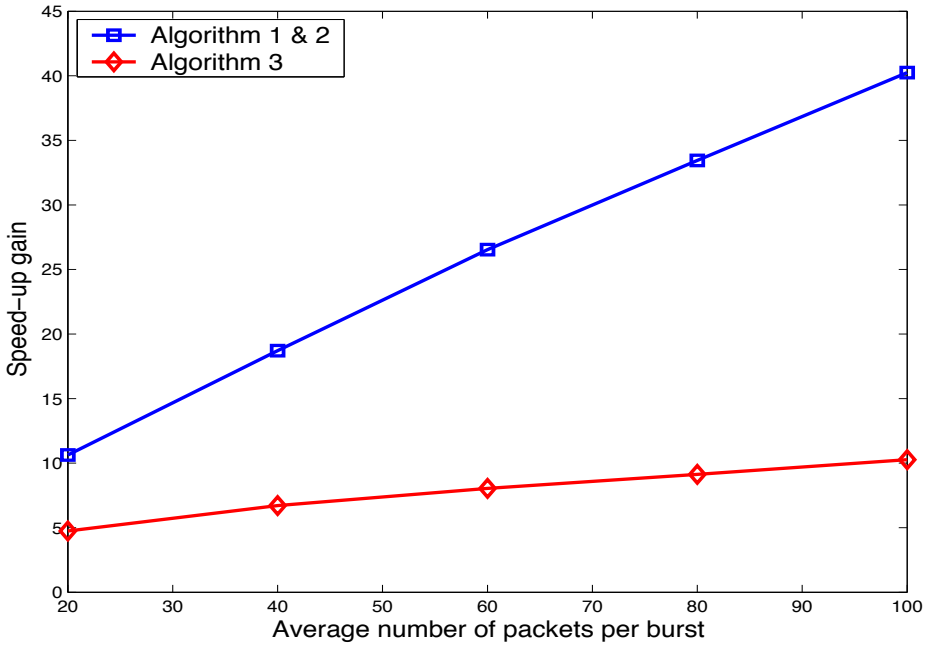

Figure 4.10: Speed-up gain of using the burst generator algorithms over direct simulating of the assembly algorithms for generating 10^7 bursts as a function of average number of packets per burst.

tion time increases with the number of packets per burst. The speed-up gain for the hybrid algorithm is also quite significant, however, it is smaller than that of the first two algorithms. In fact, the difference between speed-up gains of Algorithm 1(2) and Algorithm 3 is that in the latter case, generating the conditional probabilities requires going through a while loop that, in turn, increases the simulation time.

4.5 Performance Analysis and Discussion

In the last section we observed that the burst assembly shape the traffic that is injected into the OBS network. In this section we investigate the impacts of this shaping on the contention resolution inside the network. For this purpose, the performance of a network link fed by superposition of several burst flows is studied and the impacts of key parameters such as the assembly thresholds, the number of flows and the FDL buffer capacity are analyzed.

4.5.1 System Model

Our performance analysis is carried out on a single output port of a core switching node with W wavelength channels of rate C Gb/s. It is assumed that the store-and-forward switching paradigm is employed in the network. The node supports full wavelength conversion capability, and each output port of the node is equipped with a small-size WDM FDL buffer. The buffering unit is employed in a dedicated single-stage feed-forward architecture, see Chapter 3, and arriving bursts may undergo wavelength conversion – performed by tunable wavelength converters (TWC) – before entering the buffer, that is, every data unit that goes through the buffer makes use of the same wavelength in the FDL unit and over the outgoing link. The FDL buffer comprises $M+1$ delay lines as $(D_0, D_1, D_2, ..., D_M)$, with $D_0 = 0$ and the i^{th} delay line providing delay of $i \times D$ with $(1 \leq i \leq M)$, where D is the so-called buffer granularity.

There are N ingress traffic sources feeding the considered output port, where each source is represented by a single burst assembler. In fact, the total number of flows that can concurrently inject traffic into an output port of a core node depends both on the total number of traffic sources, i.e., ingress nodes, as well as on the size of the node itself. Nevertheless, here we assume that size of the node does not limit the number of flows. Also, we neglect the pre-transmission delay that bursts may experience in burst buffers.

4.5.2 Simulation Experiments

The model has been simulated with discrete event simulator OMNeT++ [154] by developing and incorporating additional functionalities required for OBS. In simulation results shown below, the method of batch means [153] has been used to estimate the mean values of the drop rates. We have estimated 90% confidence intervals for the results.

To evaluate the impact of traffic shaping on performance of the system, we have performed two sets of experiments. In either case the major performance metric considered is the burst drop rate.

Experiments Set I

The goal of this experiments set is to analyze the impacts of different burst assembly parameters on the burst drop rate over the link. For the analysis we make the following assumptions.

- The total number of flows being multiplexed onto the link N is fixed and equal to 10.
- Each burst flow has the data rate of 25 Mb/s. That is, the flow resulting from multiplexing burst flows has the data rate of 250 Mb/s.

- The link has 4 wavelength channels with the total capacity of 350 Mb/s. That is, the total load offered to the link is 0.71.
- The buffer granularity D is set to the average burst size in each experiment.
- Arriving data units are scheduled over the channels using the latest available unscheduled channel (LAUC) scheduling algorithm, see Chapter 3.

In the last section, it was demonstrated that different settings of assembly parameters can be well reflected in P_{Tout}. Accordingly, we devise five scenarios based on different P_{Tout} values as explained below. In scenario I, II, III and IV, all the assemblers are homogenous and run at P_{Tout}=0, 0.25, 0.5 and 0.75, respectively. In scenario V, 3 assemblers run at P_{Tout}=0.75, 4 assemblers run at P_{Tout}=0.5 and finally 3 assemblers run at P_{Tout}=0.25.

Table 4.4 shows drop rates on the link for different scenarios. The results are also compared with a baseline scenario, in which burst arrivals to the link follow the Poisson process and burst lengths are taken from the burst trace generated in Scenario I. It is seen that for a given setting, i.e. given number of available delay lines, there is no substantial difference among the loss rate values of different scenarios. This indicates that, although a single traffic flow aggregated at $P_{Tout} = 0.75$ is less bursty than the same trace aggregated at $P_{Tout} = 0$ (see Figs. 4.8 and 4.9), when several such flows are multiplexed together, the impacts of P_{Tout} on the burst drop rate is not significant. Nonetheless, observe that the results of all considered scenarios are better than the baseline scenario, implying that the aggregate traffic flow resulting from multiplexing single burst flows is considerably smoother than the Poisson flow (see Fig. 4.11).

At a given P_{Tout}, the variation of burst interdeparture times of a single burst flow is further reduced through increasing the length threshold L_{Th} as depicted in Table 4.3. Accordingly, we also evaluate the burst loss rate on the link at $L_{Th} = 20$ KB as shown in Table. 4.4. In this case, the results are only presented for Scenario I, as changing the scenario impacts the performance only marginally. It is seen that, although increasing the burst length has no impact on the loss rate of the link with no FDL buffer, it does improve the drop rate when the link is equipped with FDL buffers. More specifically, the larger the number of delay lines, the better the performance compared to the case with $L_{Th} = 10$ KB. For instance when there are 2 delay lines available, the drop rate at $L_{Th} = 20$ KB is around one order of magnitude better that that at $L_{Th} = 10$ KB.

The points to take away from this experiments set can be summarized as follows.

- the superposition of several assembled burst flows achieves a better performance – in terms of burst loss rate – than a single flow, in which burst arrivals follow the Poisson process. The difference is particularly large – in the range of orders of magnitude – when the link under consideration is equipped with FDL buffer.
- On a link with no FDL buffer and fixed number of multiplexed assembled flows, changing the assembly parameters does not result to a noticeable difference on the burst drop rate.

Table 4.4: Average burst loss rate on the link. Unless otherwise stated all constituting traffic flows are generated at $L_{Th} = 10$ KB . The values are shown with 90% confidence level.

No. FDL	Baseline	Sc. I ($L_{Th} = 20$ KB)	Sc. I	Sc. II	Sc. III	Sc. IV	Sc. V
0	0.19 $\pm 2.5 \times 10^{-4}$	0.15 $\pm$ 1.5$\times 10^{-4}$	0.15 $\pm$ 2.7$\times 10^{-4}$	0.15 $\pm 1.3 \times 10^{-4}$	0.15 $\pm 1.4 \times 10^{-4}$	0.15 $\pm 2.5 \times 10^{-4}$	0.15 $\pm$ 2$\times 10^{-4}$
1	7×10^{-2} $\pm 2.2 \times 10^{-4}$	4.77×10^{-3} $\pm 3.3 \times 10^{-5}$	8.75×10^{-3} $\pm 6 \times 10^{-5}$	8.67×10^{-3} $\pm 5.1 \times 10^{-5}$	9.37×10^{-3} $\pm 4.5 \times 10^{-5}$	1.19×10^{-2} $\pm 9 \times 10^{-5}$	1.03×10^{-2} $\pm 1.7 \times 10^{-4}$
2	3.56×10^{-2} $\pm 1.8 \times 10^{-4}$	2.49×10^{-5} $\pm 3.4 \times 10^{-6}$	3.15×10^{-4} $\pm 1.6 \times 10^{-5}$	2.74×10^{-4} $\pm 9.9 \times 10^{-6}$	3.7×10^{-4} $\pm 1.6 \times 10^{-5}$	7×10^{-4} $\pm 3.6 \times 10^{-5}$	4.59×10^{-4} $\pm 1.1 \times 10^{-5}$

- On a link with FDL buffers and fixed number of multiplexed assembled flows, the burst drop rate can be improved by increasing the length threshold of corresponding burst assemblers.

Experiments Set II

In the last experiments set, we assumed that the number of burst flows is fixed. In this experiment we investigate the impact of number of multiplexed burst flows on the performance of the system described above. For this purpose we consider the same assumptions made in the first experiments set, except that the number of multiplexing flows and the rate of the flows are not fixed in this experiment. Specifically, we vary the number of flows but we keep the total offered load to the link fixed at 0.71. In any case, all the burst flows are generated at $L_{Th} = 10$ KB and $P_{Tout} = 0.5$. The simulation results are presented in Table. 4.5.

We observe that the number of multiplexed burst flows has a large influence on the drop rate, so that the less the number of the flows, the better the performance. Although reducing the number of multiplexed flows results in the improvements of the drop rate for both with and without FDL buffers, the improvement with FDL buffer is more profound. For instance, when there are two delay lines available in the system, reducing the number of flows from 50 to 10 improves the loss rate for around two orders of magnitude. This can also be understood intuitively, by taking into account that decreasing the number of independent flows that are being multiplexed onto a single link, while keeping the cumulative offered load fixed, reduces the chances of contentions among bursts of different flows.

To further reasoning about this behavior we turn to the CoV of number of burst arrivals to the link. Fig. 4.11 depicts the CoV curves as a function of time scale for the traffic flows used in this experiment. The results are further compared against a fictitious burst flow with the Poisson arrivals at the same rate. First, it is seen that the CoV values of the multiplexed burst flows are much smaller than those of the Poisson traffic over the considered time scales, and the difference between the two decreases with the time scale. Also, the CoV values of the multiplexed flows reduces with the inverse of number of multiplexed burst flows. These results justify the sensitivity of the

Table 4.5: Average burst loss rate on the link. All constituting traffic flows are generated at $L_{Th} = 10$ KB. The values are shown with 90% confidence level.

No. FDL	Number of multiplexing flows				
	5	10	15	20	50
0	0.15 $\pm 2 \times 10^{-4}$	0.15 $\pm 1.4 \times 10^{-4}$	0.169 $\pm 2 \times 10^{-4}$	0.174 $\pm 1.7 \times 10^{-4}$	0.184 $\pm 2.1 \times 10^{-4}$
1	3.85×10^{-3} $\pm 4 \times 10^{-4}$	9.37×10^{-3} $\pm 4.5 \times 10^{-5}$	1.87×10^{-2} $\pm 1.2 \times 10^{-4}$	2.7×10^{-2} $\pm 1.5 \times 10^{-4}$	5.12×10^{-2} $\pm 1.9 \times 10^{-4}$
2	1.56×10^{-4} $\pm 1 \times 10^{-5}$	3.7×10^{-4} $\pm 1.6 \times 10^{-5}$	8.33×10^{-3} $\pm 1.3 \times 10^{-5}$	1.68×10^{-3} $\pm 4.6 \times 10^{-5}$	1.18×10^{-2} $\pm 1.1 \times 10^{-4}$

burst drop rate on the link to the number of multiplexed burst flows and also the large difference between the drop rates under Poisson traffic and assembled burst traffic.

The results achieved from these experiments lead us to the conclusion, that in order to mitigate the contentions problem in OBS networks through traffic shaping the efforts should be focused on decreasing the number of traffic flows through aggregating traffic into flows with higher capacities at the edge of the network.

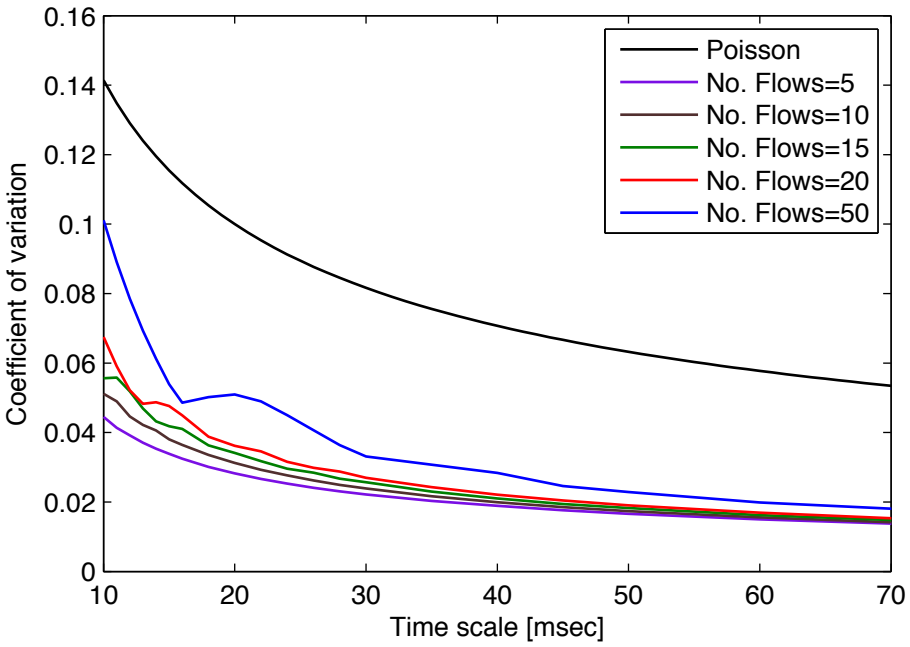

Figure 4.11: Coefficient of variation of number of burst departures as a function of time scale. All flows are aggregated at L_{Th}=10 KB and $P_{Tout} = 0.5$.

Chapter 5

Burst Assembly Process: A Trace-Based Investigation

The role of properly designing the burst assembly process is crucial with respect to the operation and performance of the OBS network. Consequently, in order to design an operational OBS network, the first step is to study the burst assembly algorithms. In Chapter 4, making some simplifying assumptions about the input packet arrivals to the assembly unit, we modeled the most common assembly algorithms and presented exact analytical formulae for the characteristics of the assembled traffic. In this chapter we put our focus on the analysis of the assembled traffic using the real measurement-based IP traces as the input to the assembly buffer. The objective of this study is twofold. First, we examine the validity of the conclusions that are drawn from our analytical study in Chapter 4 and above that, we gain a clear insight into the nature of the traffic injected into the OBS network.

5.1 Related Work

Several studies have dealt with traffic characteristics and associated performance implications inside OBS networks. The existing works in this area are in general classified into two categories. First, studies that take an analytical approach or use simulation experiments based on synthetic traffic models. The examples of the works belonging to this category are [148, 150, 149, 151, 155, 156, 88]. The second category includes simulation studies based on real traces collected from the Internet. The works presented in [157, 158, 159, 160] are the major existing literature of the second category. A review of analytical studies in [148, 150, 149, 151] are presented in Chapter 4. In the following we briefly review the results presented in the rest of the works.

Authors in [155] characterize the traffic at the output of a time-based burst assembler assuming that the input packet-level traffic can be modeled by the fractional gaussian noise (FGN) process. They also demonstrate that self-similarity does not have an adverse effect on the blocking probability of a bufferless OBS core switch, as

compared to the case where the burst traffic is assumed to be Poisson. In [156], using synthetic self-similar input traffic it is claimed that the self-similarity in the traffic injected to the OBS network can be reduced through the assembly process. However, in [88] it is demonstrated that this reduction is only related to short time scales and also depends on the load offered to the assembly buffer.

In [157] a trace-based simulation is conducted to investigate the impact of the time threshold in a timer burst assembler on the fraction of bursts that are generated with a length smaller than the minimum length requirements of the system. In [158], the effect of assembly parameter on the scaling properties of multifractal traffic is analyzed. The authors use real Internet traces as input and conclude that the key factor in characterizing the output of the assembler is the relationship between the burst assembly time scale and the cut-off time scale of the input traffic. In [159], authors use records of MPEG4 video frames to verify the analytical models that they have developed for the burst assembly process. In [160], authors study the statistical characteristics of traffic at the output of burst assembly using traces collected form the Internet. The work in [160], to our knowledge, is the only available literature that investigates the main characteristics of the burst-level traffic such as burst length and burst interdeparture time using real Internet traces. Nonetheless, a clear and thorough picture of the traffic statistics for the hybrid burst assembly is missing in this work. Furthermore, the authors ignore the classification of input IP packets according to their destination address that, as we explain in the next section, is an integral part of the burst assembly process. That is, the burst assembly algorithm is applied to all the packets traveling over a link irrespective of their destination address.

The work presented in this chapter enjoys several unique features as compared to the available literature. First and foremost, in this study we take into account the destination-based classification of packets at the input of the burst assembler. Further, we evaluate the impact of the assembly parameters and traffic intensity on the generated traffic through an intermediate utility function called P_{Tout}. More specifically, the input parameters are first mapped into P_{Tout}, which assumes values in $[0, 1]$ and gives the fraction of bursts that are generated due to the expiry of the burst assembly timer. A large set of experiments are designed and carried out to characterize P_{Tout} and its relation to the input parameters. Then, the burst level traffic is characterized at different values of P_{Tout}. Since the analysis involves several input parameters, taking this approach brings some structures into the analysis and improves its tractability.

5.2 Input Packet Traces

We focus on the hybrid burst assembly algorithm and as its input we use packet level traces collected on 18th. July 2007 from the Munich scientific network (Muenchener Wissenschaftnetz, MWN) in Germany. This network connects some 55,000 hosts at two major universities and several research institutes to the Internet via a 10 Gb/s link.

Table 5.1: Input traces used in our analysis.

Trace	no. of packets	data rate (Mb/s)	mean packet size(B)	mean packet interarrival time (μs)
TR1	17672402	24.94	645.47	206.87
TR2	12368694	17.48	645.73	295.57
TR3	8836042	12.47	645.18	413.74

The daily volume of traffic transferred over this link is around 3-6 TB. The trace of packets traveling on this link has been recorded for 24 hours, though for our analysis we select a part of the trace which is one hour long (10a.m.-11 a.m.). Over this one-hour period the link is highly utilized.

As explained in the last section, in the first step, we need to classify packets according to their destined egress edge node. The exact implementation of the classification necessitates knowledge about the exact topology as well as routing in the considered network. Alternatively, one can classify packets based on the autonomous system (AS) they are destined to. This is one of the closest approximations to the edge-based classification that is feasible without much effort. In fact, AS-based classification might be somewhat finer than the classification based on the egress edge node, therefore it is considered as a simple method mimicking the edge based classification in a conservative manner. Accordingly we opt for the AS-based classification in this work. As a result of the classification, we come up with several subtraces, each with a different destination AS. For our analysis we rely on the subtrace with the highest data rate.

Since we would like to analyze the impact of traffic intensity on the burst assembly process, we decided to downsample the selected subtrace at different rates. For this purpose, we apply the connection-level downsampling, where packets belonging to different end-to-end connections are identified based on their source and destination IP addresses and port numbers. After having identified the connections in the considered subtrace, we randomly remove a certain fraction of the connections from the trace so that the data rate of the resulting subtrace is equal to the desired data rate. In this way, we create two additional subtraces whose data rates are 70% and 50% of the original subtrace. Table 5.1 depicts the information of the three traces that are used in our analysis.

5.2.1 Statistical Characteristics of Packet Traces

One of the major distinctions between the work presented here and the other related works discussed earlier is using different types of input to the system. Therefore, it is important to first explore the statistical characteristics of the three traffic traces, which are denoted as TR1, TR2 and TR3 in Table 5.1. Fig. 5.1 shows the volume of the traffic

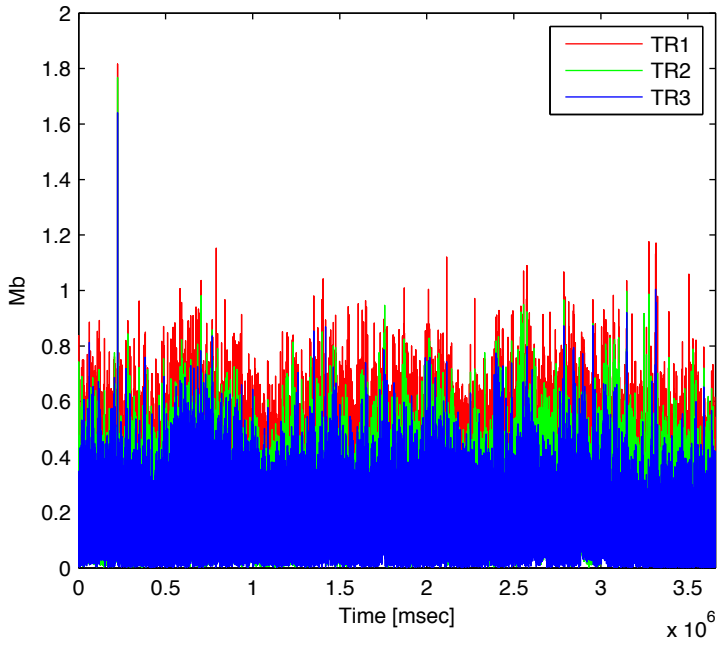

Figure 5.1: Input packet traces: volume of arriving traffic averaged over 10 ms intervals.

associated with each of the three traces averaged over 10 ms intervals. Also, Fig. 5.2 shows the empirical complementary cumulative distribution function (CCDF) of packet interarrival times for the traces. We observe that in all three cases, the plotted CCDF is a straight line, except for the very small interarrival times, which is an indication that the marginal distribution of packet interarrival times can be well approximated by the exponential distribution. In fact, the minor deviation between the CCDFs and the exponential distribution at the very small interarrival times is associated to the back to back arrival of packets on the link.

Depicted in Fig. 5.3 is the empirical cumulative distribution function (CDF) of packet sizes for the three traces. The packet size varies between 40-1500 B and the largest spikes are located at 40, 1492, 53 and 1500 B, respectively.

To investigate the correlation structure of the packet traces, we estimate the auto-correlation of packet interarrival times for the lags up to 30. As depicted in Fig. 5.4, the packet arrival is a correlated process and the three examined traces exhibit more or less similar correlation structure. To check for the scaling property in the packet traces, we conduct a wavelet-based multi resolution analysis (MRA) [161]. Fig. 5.5 depicts that there is a strong scaling behavior with the estimated Hurst parameters 0.947, 0.972 and 0.947 for TR1, TR2 and TR3, respectively. Also, it is seen that the onset of LRD for all three cases is at scale $j = 5$ corresponding to 300-500 ms.

As we explain in the next section, the short range burstiness of the input traces is a factor that can largely influence the burst-level statistics. Therefore, in Fig 5.6 we

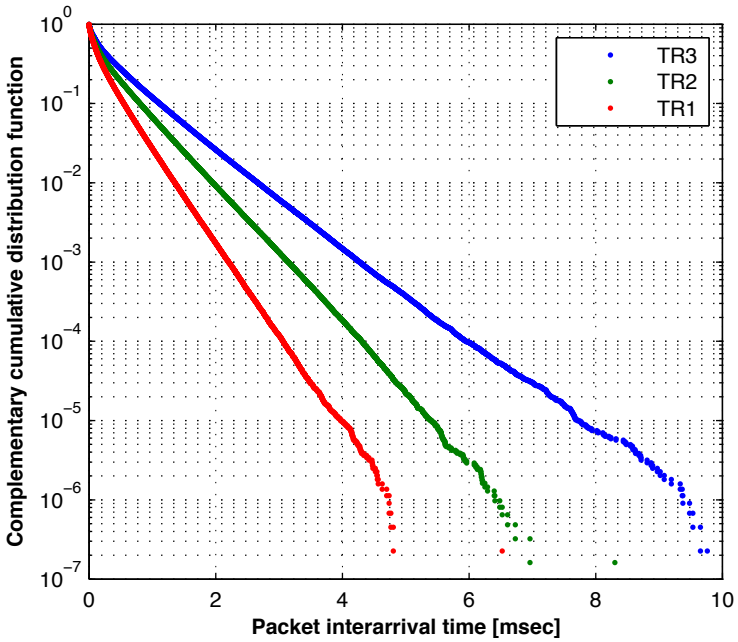

Figure 5.2: Input packet traces: CCDF of packet interarrival times.

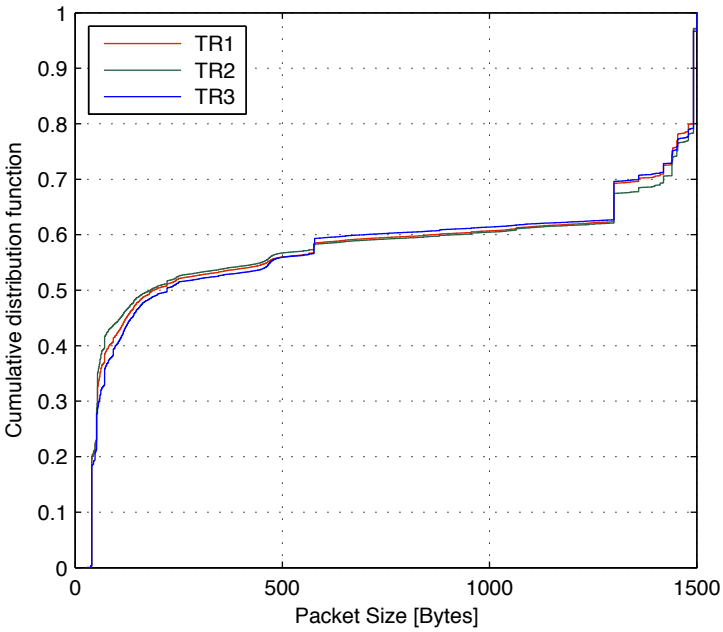

Figure 5.3: Input packet traces: CDF of packet sizes.

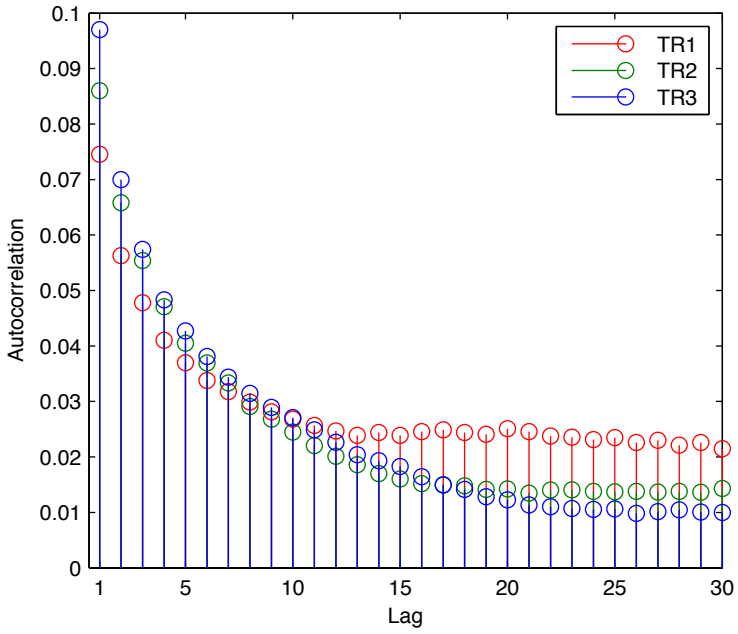

Figure 5.4: Input packet traces: autocorrelation of packet interarrival times.

estimate the coefficient of variation (CoV) of the volume of traffic collected over small (up to 50 ms) time intervals. We observe that in the short time scales the traffic gets smoother with time and also the CoV decreases with input data rate.

In summary, at the input to the assembler we have a correlated long-range-dependent process, which is quite different from that assumed in analytical works presented in Chapter 4. More precisely, although the marginal distribution of the packet interarrival time follows the exponential distribution and is the same as the assumption we make in Chapter 4, the packet size distribution and the correlation structure of the arrival process are clearly different from our assumptions in last chapter.

5.3 Burst-Level Analysis

In this section we analyze the characteristics of traffic at output of the burst assembly unit. For this purpose we develop a simulation model of the hybrid burst assembly algorithm, as the most general algorithm, in Perl [162]. The model allows us to apply the burst assembly to the recorded packet traces that were discussed in the previous section. The output of the simulation model are traces of bursts that are recorded for further statistical evaluations. In order to characterize the burst-level traffic, we should conduct several simulation experiments while varying different parameters of the system. There are three important such parameters, namely input traffic rate, the time threshold T_{Th} and the length threshold L_{Th}. Therefore, we must design our

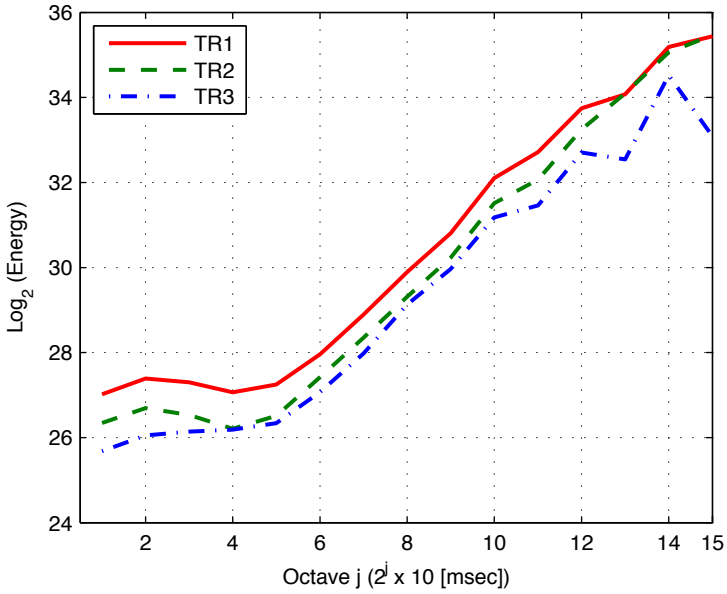

Figure 5.5: Input packet traces: wavelet-based global scaling plot using wavelet Daubechies 4 (byte wise).

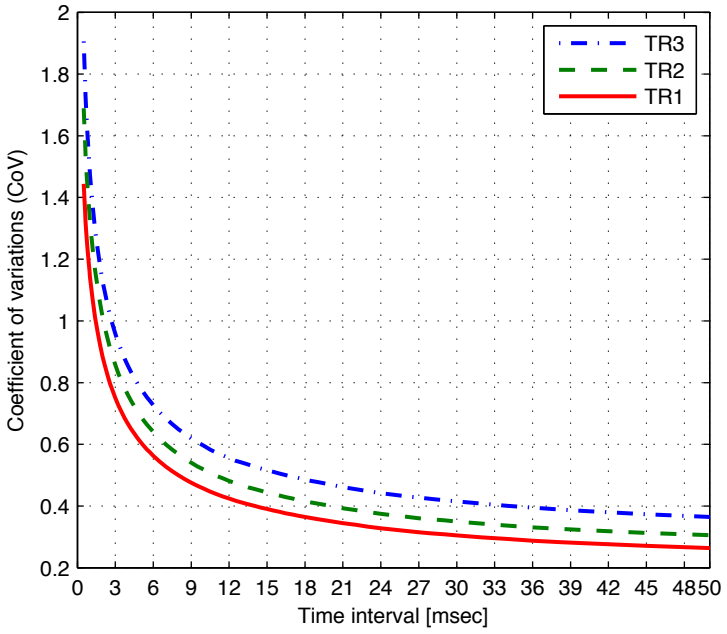

Figure 5.6: Input packet traces: coefficient of variation of number of arriving bytes as a function of time scale.

experiments in a way that we can observe the impact of all these parameters on the burst-level characteristics. On the other hand, from the discussion in Section 4, we know that depending on the combination of these three parameters there might be two types of bursts at the output of the burst assembler. Namely, bursts that are generated due to expiration of the assembly timer (hereinafter type T bursts) and bursts that are generated before the timer expires because the length threshold is met (hereinafter type Z bursts). Consequently, the statistical characteristics of generated burst trace will be largely related to the relative fraction of type T (Z) bursts, i.e., the probability that the assembly timer goes off, P_{Tout}. In order to facilitate the process of characterizing the burst traffic, we first characterize P_{Tout} and investigate the impact of input parameters on this function. Then, in the next step, we design our experiments based on different values of P_{Tout}.

5.3.1 Analysis of P_{Tout}

Let us begin our analysis with an illustrative example and assume that the input packet-level traffic is of type constant bit rate (CBR) with rate R_{CBR}. For given values of input traffic rate and the length threshold, P_{Tout} as a function of T_{Th} has a deterministic behavior as depicted in Fig. 5.7. That is, there is a certain value of t^* that if the time threshold is set to be smaller than this value, all generated bursts will be of type T, otherwise all of them will be of type Z. Now, let us replace the CBR packet trace at the input with a bursty trace like the ones we consider in our experiments. In this case, P_{Tout} will no longer be a deterministic function of P_{Tout}, however, we still can argue that the assembly process is a renewal process. That is, the assembly buffer forgets its history every time a new burst is generated and sent out. On account of this, the deviation of P_{Tout} from the one depicted in Fig. 5.7 will only be influenced by the variation of input traffic at time scales not greater than the timer threshold of the assembly process. As a measure of the variation of the input traffic we rely on the CoV of the input traces as illustrated in Fig 5.6.

To characterize P_{Tout}, we use the simulation model to estimate the fraction of T bursts at different values of time and length threshold for all the three packet traces. The simulation results are shown in Figs. 5.8 and 5.9. Furthermore, and for the sake of comparison, we calculate values of P_{Tout} as a function of T_{Th} based on the analytical models developed in Chapter 4 for TR1. The results from the analytical model are denoted as *Model* in Fig. 5.10 and compared to the results of our simulation, denoted as *Sim.* Note that for calculating P_{Tout} using the analytical model, we only need the mean input traffic rate, average packet size – which are both easily extracted from TR1 – and the assembly parameters. We observe that there is a clear distinction between results of the analytical model and the simulation, which stems from the difference in the input traffic characteristics of the two cases. Although the model accurately predicts the value of T_{Th} at which $P_{Tout} = 0.5$, the range of variation of T_{Th} when P_{Tout} varies from 0 to 1 is much larger for the simulation than the analytical case.

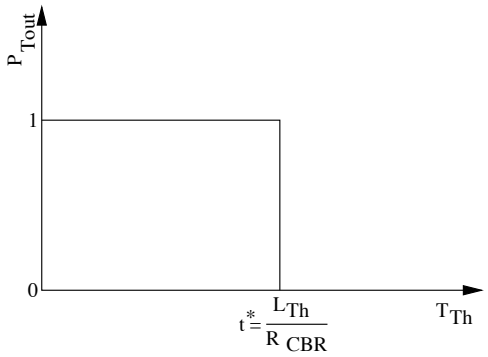

Figure 5.7: Probability of the assembly timer expiry as a function of T_{Th} at given L_{Th} and input traffic rate for the CBR traffic.

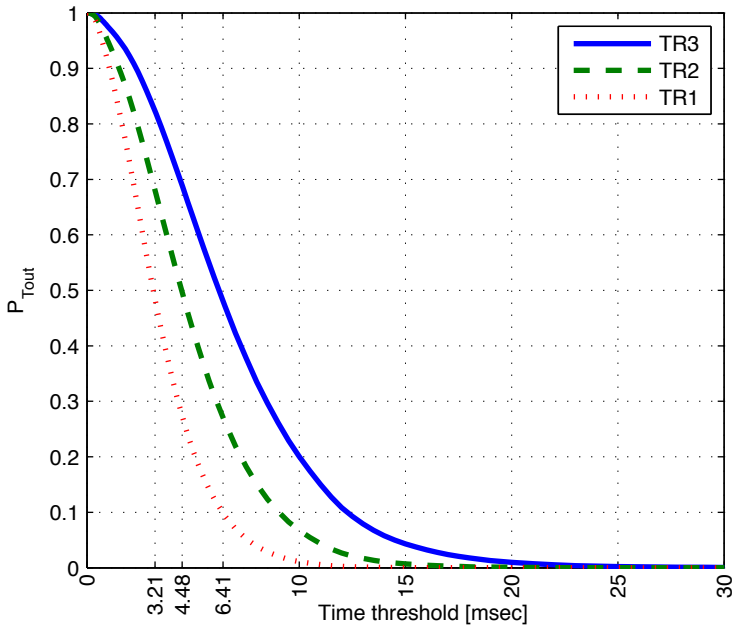

Figure 5.8: Probability of the assembly timer expiry with the packet traces as input as a function of T_{Th} at L_{Th}=10 KB.

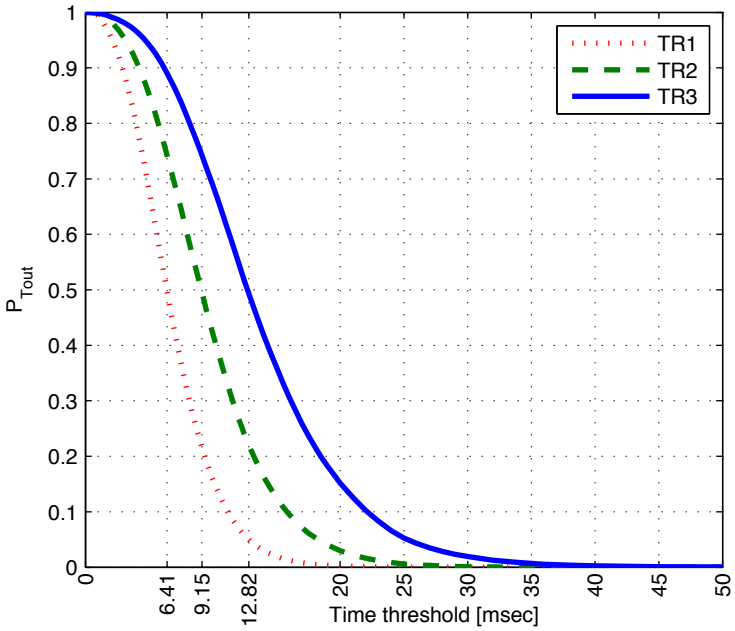

Figure 5.9: Probability of the assembly timer expiry with the packet traces as input as a function of T_{Th} at L_{Th}=20 KB.

Now let use further explore the results of our experiments. From Figs. 5.8 and 5.9 we observe that for given input traffic rate R and length threshold L_{Th}, the value of time threshold at which $P_{Tout} = 0.5$ can be very well approximated by $\frac{L_{Th}}{R}$. The labels of the x-axis on plots of Figs. 5.8 and 5.9 at (3.21, 4.48, 6.41, 9.15, 12.82, 16.04, 32.07) depict $T_{Th} = \frac{L_{Th}}{R}$ for the corresponding settings. To evaluate the accuracy of this approximation, we calculate the relative error between the value of T_{Th} that results in $P_{Tout} = 0.5$ in our simulation and its corresponding approximation in Table 5.2. The accuracy of this approximation improves with the input traffic rate and the selected length threshold so that the relative error is less than 1% for settings with $L_{Th} \geq 20$ KB at all three input data rates. To explain this behavior, we notice that for a given input rate, the corresponding time threshold leading to $P_{Tout} = 0.5$ increases with the length threshold that, in turn, leads to a smaller CoV according to Fig. 5.6. When the CoV decreases, we expect the traffic rate to be closer to the average value and hence a better approximation. We further note that increasing L_{Th} from 10 KB to 20 KB has a very large impact on the relative error, though this is not the case when we further increase L_{Th} to 50 KB and 100 KB. In fact, although the CoV curve is a monotonically decreasing function of the time interval, the tilt of the curve decreases with the time interval as depicted in Fig. 5.6.

Another important observation is that for given values of L_{Th} and P_{Tout}, the input

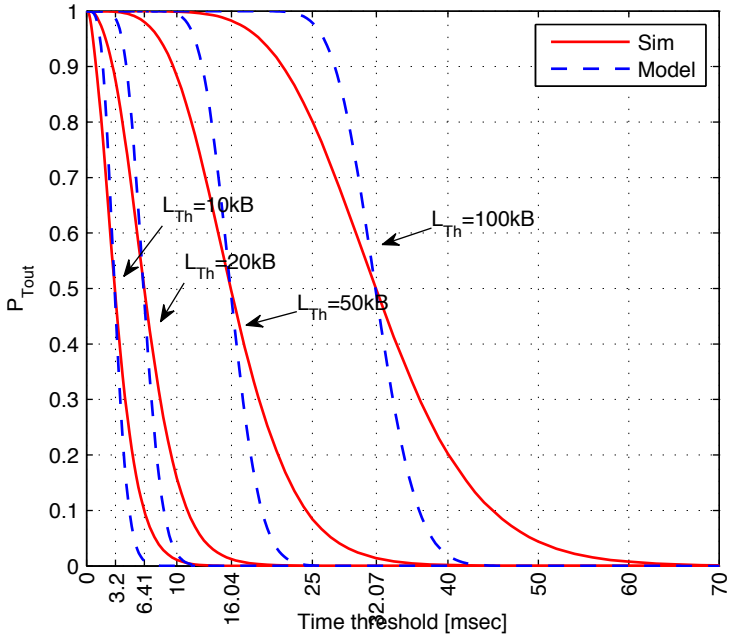

Figure 5.10: Comparison of probability of the assembly timer expiry at TR1 with analytical models.

Table 5.2: Relative error between the estimated T_{Th} at $P_{Tout} = 0.5$ from simulation and the approximation using $\frac{L_{Th}}{R}$ (the values are in %).

L_{Th} (KB)	TR1	TR2	TR3
10	2.54	2.64	3.02
20	0.45	0.68	0.82
50	0.48	–	–
100	0.17	–	–

Table 5.3: Root mean square error of estimating P_{Tout} at TR2 and TR3 based on P_{Tout} at TR1.

L_{Th} (KB)	TR2	TR3
10	1.8×10^{-3}	7.4×10^{-3}
20	2.9×10^{-3}	7.0×10^{-3}
50	1.5×10^{-3}	3.7×10^{-3}

traffic rate and the the inverse of the corresponding T_{Th} are linearly related. That is, at a given L_{Th} if the input rate increases by a factor of 2, then T_{Th} should be halved in order that P_{Tout} remains unchanged. This again is explained with the help of CoV curves in Fig. 5.6. That is, the value of CoV remains almost fixed if we scale the traffic rate and $\frac{1}{T_{Th}}$ by the same factor. Nevertheless, it is obvious that CoV does change at a given input rate, when simultaneously scaling T_{Th} and L_{Th}, meaning that there is no linear relation between L_{Th} and T_{Th} at a given input rate.

To further elaborate on the linear relationship between the input rate and the inverse of the time threshold, we take the curves of P_{Tout} versus T_{Th} for the case of TR1 at L_{Th} =10, 20 and 50 KB, and using these three curves estimate the corresponding P_{Tout} versus T_{Th} curves for TR2 and TR3 by appropriately varying the values of T_{Th}. Then for each case, we compare the results and quantify the accuracy of the estimation by calculating the root mean square error (RMSE) between the curve achieved from the simulation and the approximated one. The results shown in Table 5.3 indicate the high accuracy of the estimation.

In the next sections we focus on the characteristics of the burst traces such as burst length and burst interdeparture time distributions. For this purpose, we design several scenarios based on the three input traces, two values of L_{Th}=10, 20 KB and different values of T_{Th}. The selection of the values of T_{Th} is done based on the P_{Tout} curves of Figs. 5.8–5.10. That is, for each combination of input traffic rate and L_{Th}, we select four different values for T_{Th} in a way that they lead to $P_{Tout} = 0, 0.25, 0.5, 075$. In this way, we end up with 24 different scenarios. The corresponding values of T_{Th} for each of the scenarios are shown in Table 5.4.

5.3.2 Burst Length

In this section we characterize the marginal distribution of burst size. For this purpose we simulate the burst assembly for each of the 24 scenarios listed in Table 5.4. The corresponding CDFs of burst size are depicted in Figs. 5.11 and 5.12. Additionally, the average and variance of burst size for each scenario is shown in Table 5.5.

We observe that the CDF of the burst size is composed of two regions; a fraction equal to P_{Tout} of bursts are smaller than L_{Th} in size (type T) and the rest are those

Table 5.4: Values of T_{Th} for the 24 scenarios considered for simulation experiments (ms).

	L_{Th} = 10 KB				L_{Th} = 20 KB			
	$P_{Tout} = 0$	$P_{Tout} = 0.25$	$P_{Tout} = 0.5$	$P_{Tout} = 0.75$	$P_{Tout} = 0$	$P_{Tout} = 0.25$	$P_{Tout} = 0.5$	$P_{Tout} = 0.75$
TR1	∞	4.668	3.126	1.907	∞	8.712	6.386	4.420
TR2	∞	6.632	4.456	2.726	∞	12.310	9.088	6.318
TR3	∞	9.176	6.219	3.925	∞	17.200	12.720	9.024

Table 5.5: Estimated mean and variance of burst size for different scenarios.

		$P_{Tout} = 0$		$P_{Tout} = 0.25$		$P_{Tout} = 0.5$		$P_{Tout} = 0.75$	
		Mean(B)	Var(B^2)	Mean(B)	Var(B^2)	Mean(B)	Var(B^2)	Mean(B)	Var(B^2)
L_{Th} = 10 KB	TR1	10625	1.78×10^5	9668	4.1×10^6	8254	8.79×10^6	6147	1.19×10^7
	TR2	10635	1.78×10^5	9684	4.03×10^6	8269	8.74×10^6	6176	1.19×10^7
	TR3	10631	1.81×10^5	9704	3.85×10^6	8336	8.34×10^6	6328	1.14×10^7
L_{Th} = 20 KB	TR1	20653	1.88×10^5	19219	9.66×10^6	16979	2.25×10^7	13447	3.37×10^7
	TR2	20660	1.89×10^5	19232	9.51×10^6	17014	2.21×10^7	13533	3.32×10^7
	TR3	20648	1.9×10^5	19232	9.46×10^6	17101	2.11×10^7	13772	3.12×10^7

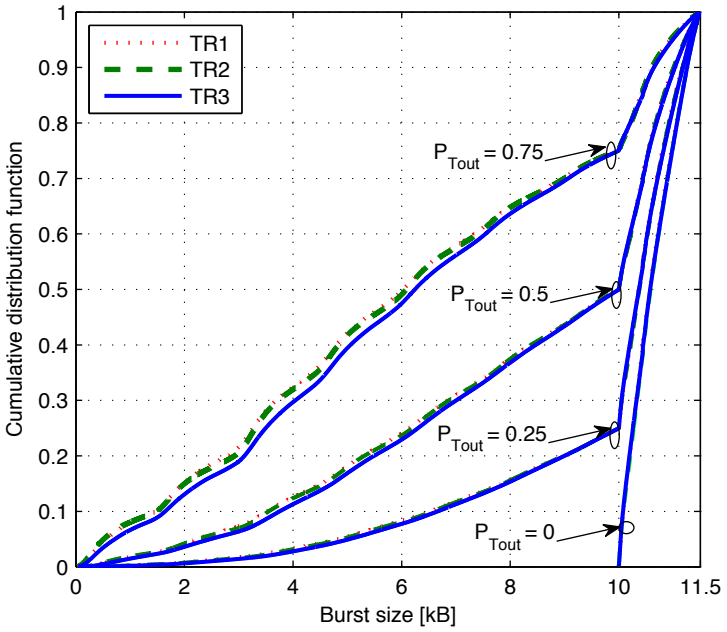

Figure 5.11: CDF of burst size at L_{Th}=10 KB.

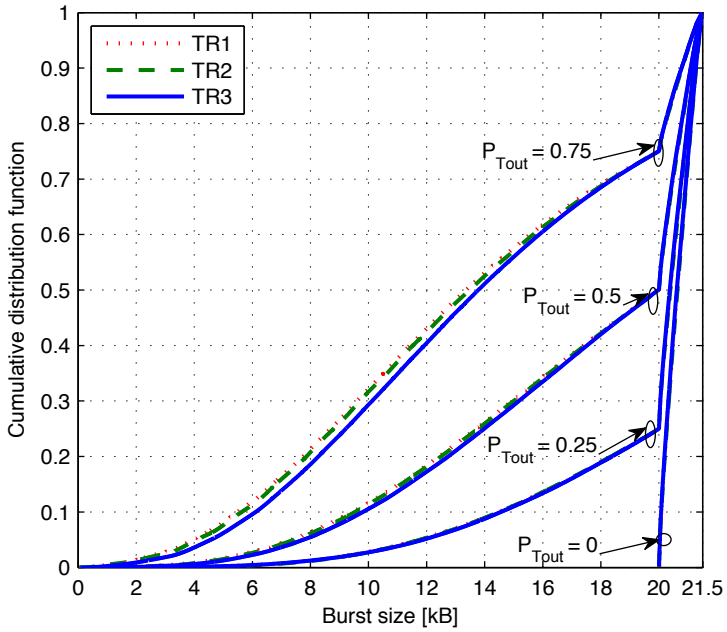

Figure 5.12: CDF of burst size at L_{Th}=20 KB.

with a size larger than L_{Th} (type Z). The distribution of type Z bursts is limited to the range $(L_{Th}, L_{Th}+1500)B$ and is influenced much by the distribution of a single packet size, which is at most 1500 B long.

For a given length threshold and input data rate, the average burst size decreases and its variance increases with P_{Tout}. There are two reasons for this. First, the average size of T bursts are smaller than those of Z bursts. In addition to that, even when we compare the average burst size only for T bursts we notice the same trend, which stems from the fact that according to Table 5.4 a decrease in P_{Tout} – caused by an increase in T_{Th} – allows the burst assembler collect more packets before the timer goes off. For the same reason, we observe that the average size of the type T bursts in Fig. 5.12 is larger than two times the average size of type T bursts in Fig. 5.11 although the length threshold only doubles.

To further study the impact of input traffic rate we consider the difference between burst sizes at fixed values of L_{Th} and P_{Tout} and at different input rates. In this case, the shape of the CDF function remains almost unchanged, however, the average size of bursts (particularly of T bursts) increases as the input rate decreases, though the difference is not much. The reason for this difference is attributed to the small difference in the variation of traffic at different rates over small time scales. Accordingly, this difference diminishes as the time scale increases.

5.3.3 Burst Interdeparture Time

Now let us consider the marginal distribution of burst interdeparture times. We use the same 24 scenarios described in Section 6.1. The CDF of burst interdeparture times are presented in Figs. 5.13, 5.14 and 5.15. Also, the mean and variance of the interdeparture times are listed in Table 5.7.

Similar to the burst size CDF, here again the CDF functions consists of two distinctive regions, namely, region I associated with the bursts whose interdeparture time is smaller than T_{Th}, and region II associated with the rest of the bursts. Note that for the case $P_{Tout} = 0$ we only observe region I since $T_{Th} = \infty$.

Recalling from Section 4, the burst interdeparture time is composed of two parts. The first part, t_p, is the difference between the time that last burst has been created and the time that a new packet arrives to the assembly buffer and triggers formation of a new burst. This part is characterized by the interarrival time of individual packets, and as demonstrated before it is well approximated by the exponential distribution. The second part of a burst interdeparture time, t_b, is characterized by the difference between the time the assembly timer has been set and the time that either the assembly timer expires or the volume criterion is met. Therefore, this part always assumes values not larger than T_{Th}.

Consequently, the region I in the CDFs of Figs. 5.13 – 5.15 is associated to the fraction of type Z bursts whose total formation time $(t_p + t_b)$ is smaller than T_{Th}. In other words, the burst interdeparture time in this region is characterized by the

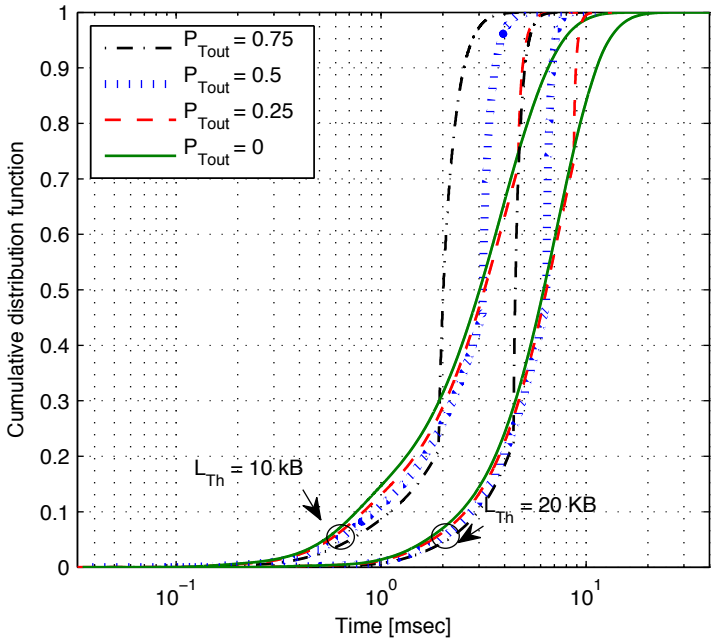

Figure 5.13: Marginal CDF of burst interdeparture times for TR1.

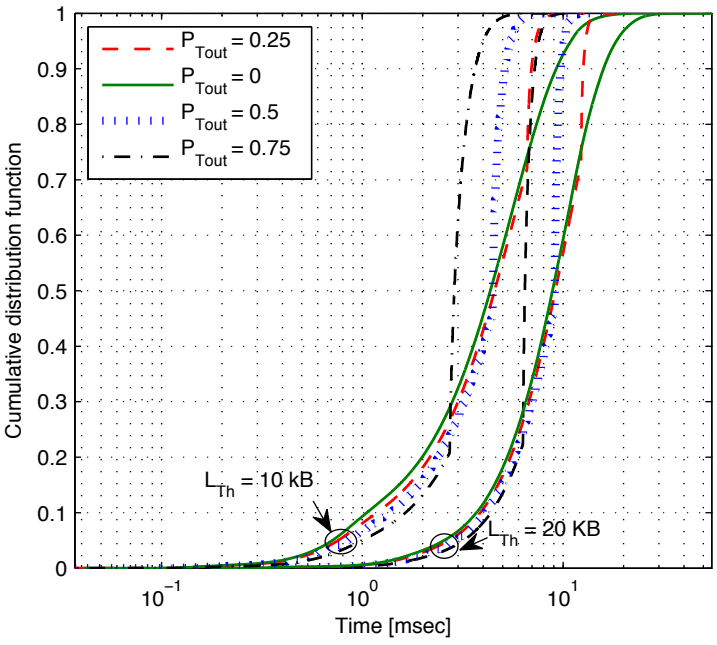

Figure 5.14: Marginal CDF of burst interdeparture times for TR2.

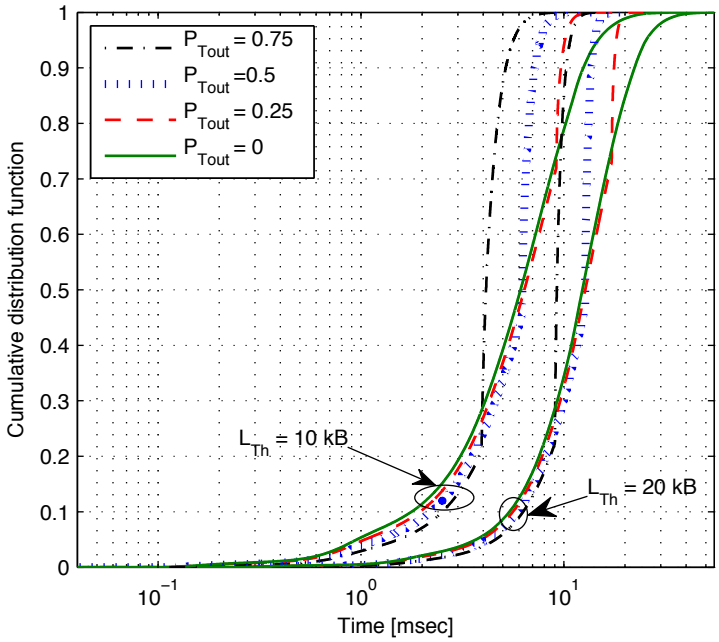

Figure 5.15: Marginal CDF of burst interdeparture times for TR3.

conditional distribution of sum of n consecutive packet interarrival times, given that this sum is smaller than T_{Th} and that the aggregate sum of corresponding packets' sizes is larger than L_{Th}. In Chapter 4, we demonstrate that for the simplified case of Poisson packet arrivals and exponential packet size distribution, this conditional distribution is characterized by the Erlang-k distribution. Similarly, from Figs. 5.13 – 5.15 we observe that the distribution in region I pictorially resembles the Gamma distribution[1], which is the general case of the Erlang-k distribution. To validate this hypothesis, we apply the nonlinear regression to the estimated CDFs of Figs. 5.13 – 5.15 for the case $P_{Tout} = 0$. In this method, the Gamma distribution is fitted to the estimated CDFs using the least-square fit method [163]. The results of regressions are shown in Figs. 5.16 and 5.17, where estimated CDFs and the corresponding best Gamma fits are plotted together. Also, Table 5.6 depicts the shape and scale parameters of the corresponding fitted Gamma distribution, denoted as a and b respectively, together with the corresponding RMSE and R^2, which serve as measures of the accuracy of the regression [163]. We observe, both pictorially and quantitatively, that the distribution

[1]The Gamma cumulative distribution function is defined as follows.

$$F(x;a,b) = \frac{1}{b^a \Gamma(a)} \int_0^x t^{a-1} e^{\frac{-t}{b}} dt \tag{5.1}$$

where, $\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt$.

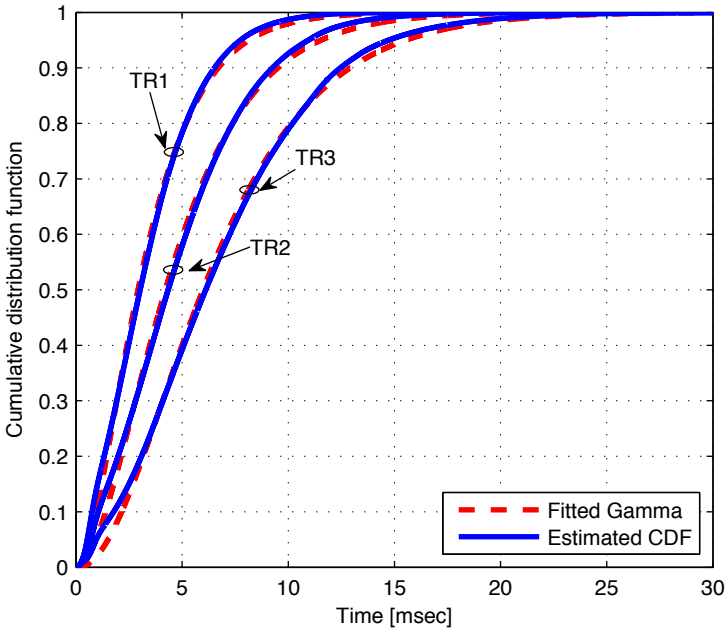

Figure 5.16: Fitting the Gamma distribution to the estimated burst interdeparture times at P_{Tout}=0 and L_{Th}=10 KB.

of burst interdeparture times in region I is consistent with the Gamma distribution and that the accuracy of the fit increases as the total number of packets contributing to the interarrival times increases.

Now we consider the region II of CDFs shown in Figs. 5.13 – 5.15. Unlike the region I, the CDF in this region is associated to both types of bursts. In fact, there might be some Z bursts whose total formation time exceeds T_{Th} if the associated t_p is large enough. Nonetheless, the fraction of these kind of bursts in region II is small in comparison to bursts of type T. In fact, for each scenario the fraction of type Z bursts that fit in region II is calculated as $1 - P_{Tout} - P_0$, where P_0 is the fraction of bursts whose interdeparture times fit in region I. From the figures it is seen that this fraction becomes smaller as the input rate increases. Accordingly, we argue that the CDFs in region II are mainly characterized by $T_{Th} + t_p$, where t_p is consistent with the exponential distribution as characterized in Fig. 5.2.

5.3.4 Burst Assembly Delay

In this section, we look into the delay that individual packets experience in the assembly buffer. The mean and variance of delay values for the considered scenarios are listed in Table 5.8. The distribution functions corresponding to TR1 and TR3 are shown in Figs. 5.18 and 5.19. The distribution for TR2 is not shown because of space limit.

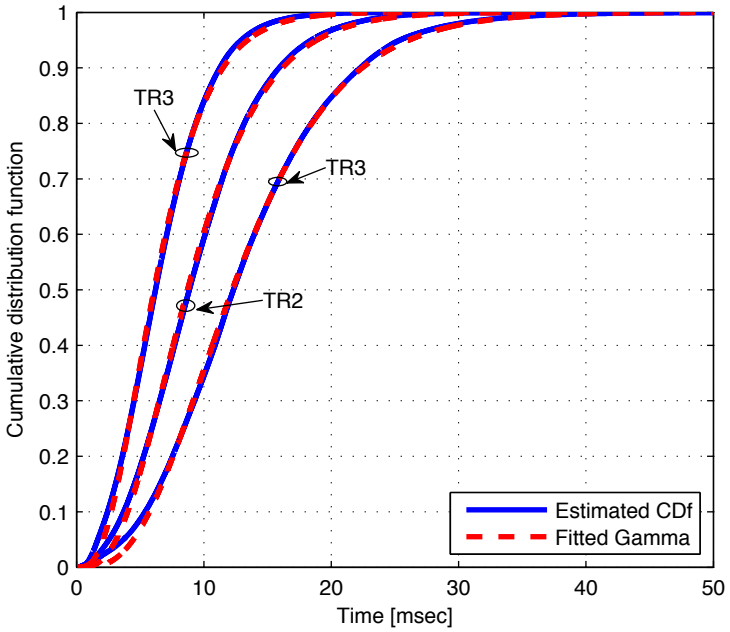

Figure 5.17: Fitting the Gamma distribution to the estimated burst interdeparture times at P_{Tout}=0 and L_{Th}=20 KB.

Table 5.6: Estimated parameters of the best Gamma fit and corresponding regression errors for P_{Tout}=0.

		(a,b)	R^2	RMSE
$L_{Th} = 10$ KB	TR1	(2.05,1.69)	.9989	8.58×10^{-3}
	TR2	(2.13,2.33)	.9988	7.5×10^{-3}
	TR3	(2.39,2.90)	.9991	7.64×10^{-3}
$L_{Th} = 20$ KB	TR1	(3.53,1.89)	.9994	7.11×10^{-3}
	TR2	(3.74,2.55)	.9996	5.89×10^{-3}
	TR3	(3.88,3.44)	.9997	5.6×10^{-3}

Table 5.7: Estimated mean and variance of burst interdeparture times for different scenarios.

		$P_{Tout} = 0$		$P_{Tout} = 0.25$		$P_{Tout} = 0.5$		$P_{Tout} = 0.75$	
		Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)
$L_{Th} = 10$ KB	TR1	3.41	5.3	3.1	2.43	2.65	1.05	1.97	3.25
	TR2	4.87	1.05	4.43	4.84	3.78	2.09	2.83	6.67
	TR3	6.82	1.92	6.22	8.88	5.34	3.84	4.06	1.28
$L_{Th} = 20$ KB	TR1	6.62	1.22	6.16	6.04	5.45	2.74	4.31	8.28
	TR2	9.46	2.39	8.8	1.17	7.79	5.36	6.19	1.64
	TR3	1.32	4.64	1.23	2.24	1.1	1.03	8.83	3.36

Table 5.8: Estimated mean and variance of packet delay in assembly buffer for different scenarios.

		$P_{Tout} = 0$		$P_{Tout} = 0.25$		$P_{Tout} = 0.5$		$P_{Tout} = 0.75$	
		Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)	Mean(ms)	Var(ms^2)
$L_{Th} = 10$ KB	TR1	1.97	3.85	1.63	1.83	1.32	9.69	0.936	3.91
	TR2	2.80	7.59	2.33	3.70	1.88	1.99	1.34	8.12
	TR3	3.88	1.43	3.25	6.94	2.65	3.78	1.94	1.65
$L_{Th} = 20$ KB	TR1	3.66	9.94	3.20	5.81	2.73	3.62	2.11	1.87
	TR2	5.19	1.98	4.55	1.16	3.90	7.31	3.03	3.81
	TR3	7.25	3.84	6.37	2.21	5.48	1.41	4.35	7.65

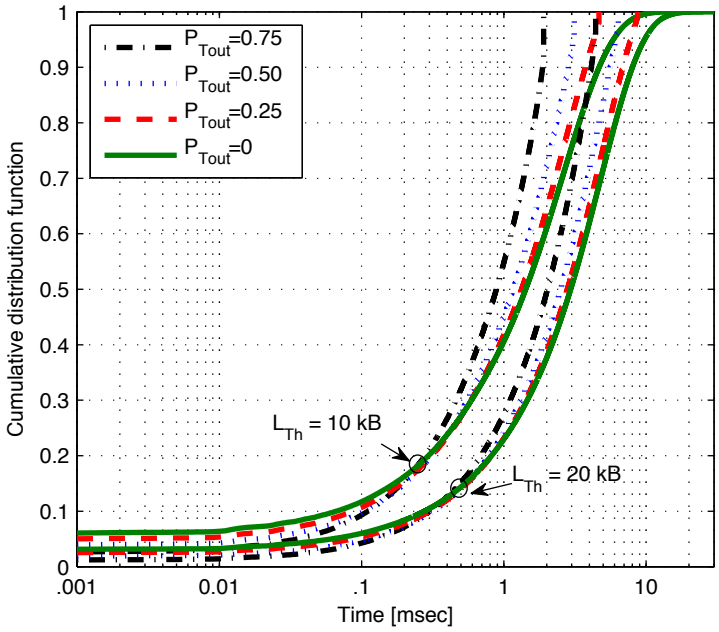

Figure 5.18: CDF of individual packet delay in the assembly buffer for TR1.

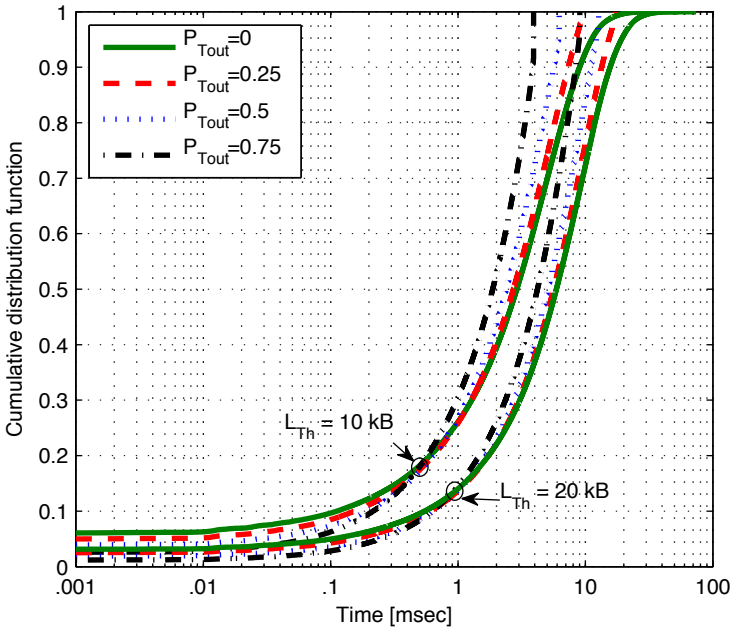

Figure 5.19: CDF of individual packet delay in the assembly buffer for TR3.

It is seen that, as expected, the average delay increases with the length threshold and the inverse of the input traffic rate. Also, for given length threshold and input traffic rate, the average delay is in direct relation with the time threshold T_{Th}. In the considered scenarios, the average value of delay is always smaller than 10 ms. Nevertheless, when we set $T_{Th} = \infty$ the maximum per packet delay increases to 73.74 ms for the scenario with TR2 and $L_{Th} = 20$ KB, though not shown here.

From the CDFs we observe that the per packet assembly delay is distributed over three regions, namely, very small delay values (from 0 up to tens of μs), delay values in the range of tens of μs up to T_{Th} and finally there is a fraction of packets with delay equal to T_{Th} (except for the case of $T_{Th} = \infty$). The very small delay values in the first region are associated with the packets that are very close to the tail of the associated bursts, i.e., the packets that arrive shortly before the timer expires in T bursts or the length threshold is reached in Z bursts. It is seen that only a small fraction of packets belong to this category, in particular, when the length threshold increases. The mass of the distribution is concentrated in the second region so that this region accounts for at least 80% of the assembled packets. The distribution of the packet delay in this region is also influenced by P_{Tout} so that when P_{Tout} approaches 1, the distribution tends towards the uniform distribution and on the other hand when P_{Tout} approaches 0 this distribution tends towards the exponential distribution. Finally, for the cases with a limited time threshold, a small fraction of packets experience a delay equal to T_{Th}. This fraction is actually related to the packets at the head of the burst. Consequently, at fixed input rate and L_{Th} the fraction of this kind of packets increases with $P_{Tout.}$

5.3.5 Number of Packets per Burst

Now let us turn to the statistics of number of packets per burst. Table 5.9 shows the main statistical measures of number of packets per burst as estimated for the scenarios introduced earlier. Also, Fig. 5.20 depicts the frequency plots of the number of packets per burst for TR1 and L_{Th}=10 KB. Due to the space limit we decided not show the frequency plots for other scenarios, whose characteristics are consistent with those of TR1.

First, we look into the impact of increasing T_{Th} while keeping the input traffic rate and the length threshold fixed. In this case, we expect that the average value of number of packets per burst increases, which is indeed the case as shown in Table 5.9. Nevertheless, this relation is not a linear one as the burst size is always limited by the length threshold.

Furthermore, the impact of changing the input traffic rate while keeping P_{Tout} fixed on the average number of packets per burst is almost negligible. Nevertheless, decreasing the traffic rate slightly increases the variance of number of packets, which is attributed to the fact that the traces with smaller rates exhibit higher variations.

We also observe that the mass of the frequency plot is concentrated around the average value so that the 98th percentile of the distribution is located at or smaller

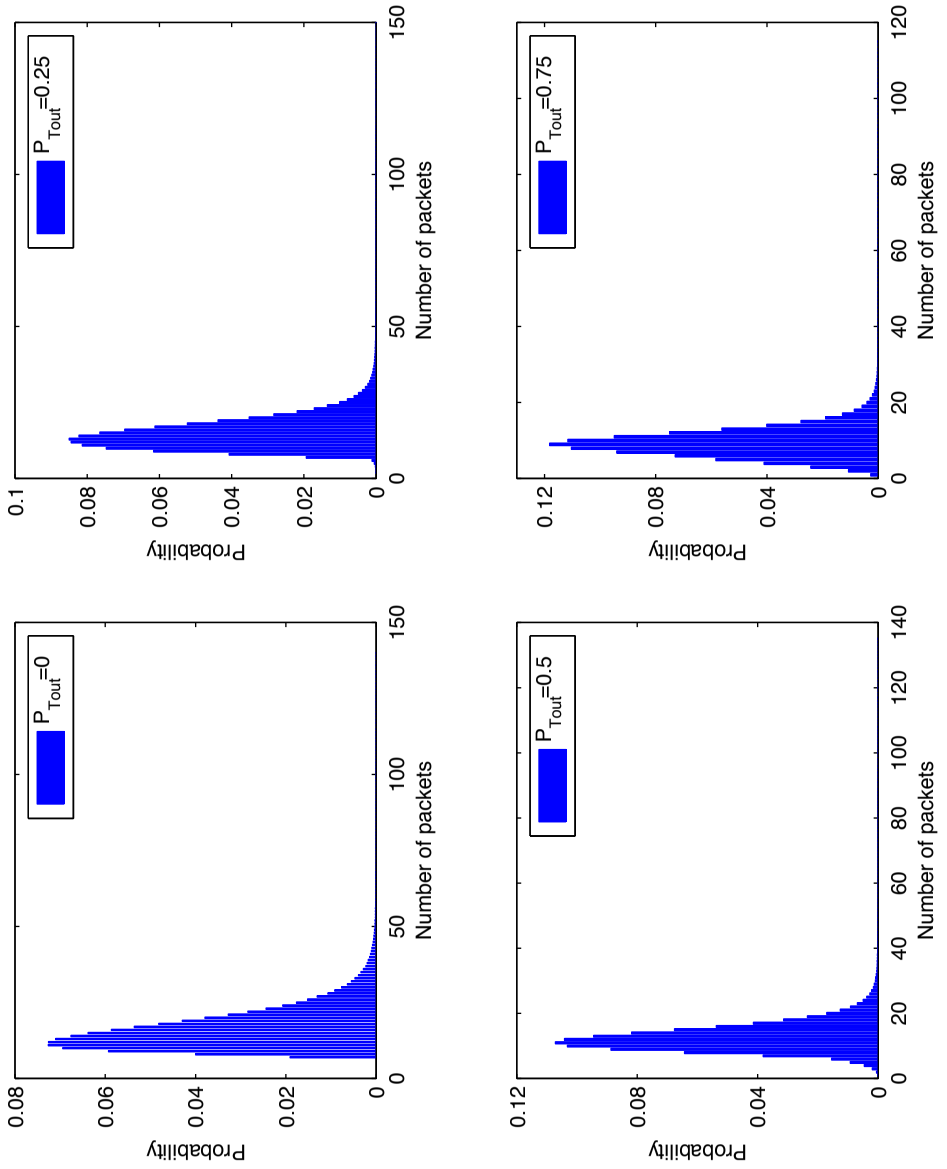

Figure 5.20: Frequency plots of number of packets per burst at TR1 and L_{Th}=10 KB.

Table 5.9: Estimated mean and variance of number of packets per burst for different scenarios.

		$P_{Tout} = 0$		$P_{Tout} = 0.25$		$P_{Tout} = 0.5$		$P_{Tout} = 0.75$	
		Mean	Var	Mean	Var	Mean	Var	Mean	Var
$L_{Th} = 10$ KB	TR1	16.47	51.90	14.99	30.52	12.80	21.33	9.53	15.72
	TR2	16.47	53.73	14.99	33.26	12.81	23.57	9.56	17.08
	TR3	16.48	54.12	15.04	34.78	12.92	25.66	9.81	18.67
$L_{Th} = 20$ KB	TR1	32.02	123.22	29.80	77.01	26.32	57.41	20.85	46.94
	TR2	31.99	129.44	29.78	85.35	26.35	65.03	20.96	52.62
	TR3	32.00	139.30	29.81	94.70	26.51	74.32	21.35	59.05

than $2 \times \bar{n}$, $\bar{n}$ being the average number of packets per burst.

5.3.6 Smoothing Impact of Assembly

Finally, we investigate the impact of burst assembly process on the burstiness of the traffic. For this purpose, we first apply the MRA method to the burst traces at the output of the burst assembler. The results for TR1 at L_{Th} =10 KB are depicted in energy plots of Fig. 5.21. To compare with the input traces, we further show the associated energy plots of the input packet trace in the same figure. It is seen that the LRD property of the input traces remains untouched by the assembly process. This, however, comes at no surprise since the impact of the assembly process on the input traffic is only limited to the time scales in the range of T_{Th} (or the maximum burst assembly time for the case of $P_{Tout} = 0$), which is always much smaller than the time scale associated with the onset of LRD. Nevertheless, Fig. 5.21 suggests that the burst assembly does reduce the burstiness of traffic at small time scales. To further investigate this, we estimate CoV of number of burst departures over different time intervals for TR1 as depicted in Figs. 5.22 and 5.23. For the sake of comparison, for each scenario we also plot the CoV of a fictitious burst arrival process that follows the Poisson process and has the same rate as the real burst trace of the considered scenario, denoted in the figures as *Poisson*.

It is seen that the burst assembly can greatly smooth out the traffic at the small time scales such that the resulting burst trace is even smoother than the Poisson process. In fact, this smoothing impact improves with P_{Tout}, which is attributed to the fact that the variability of the burst interdeparture time decreases as the fraction of T bursts increases. We further observe that scaling the burst length threshold influences the range of time scales over which the burst trace is smoother than Poisson. In fact, at a fixed input rate when the length threshold increases, the time threshold should be scaled accordingly, and this results in an increase in the time scale over which the input

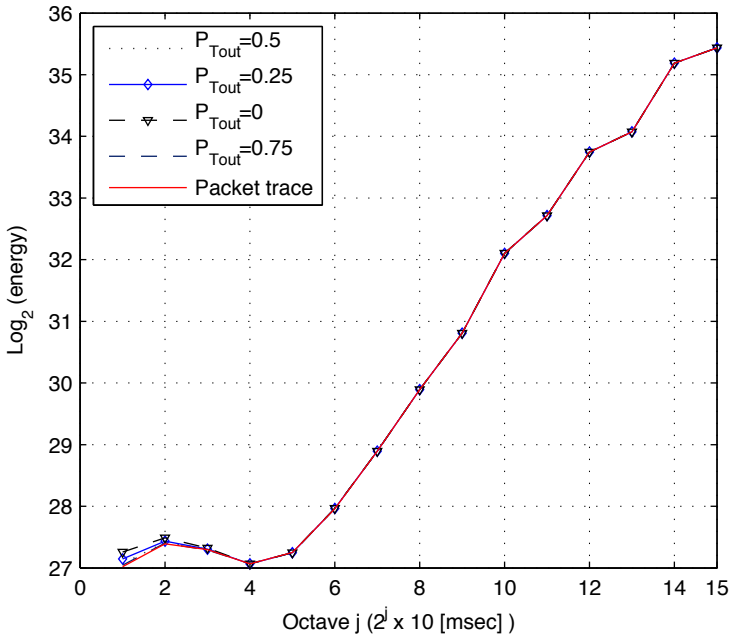

Figure 5.21: Wavelet-based global scaling plot using wavelet Daubechies 4 (byte wise) for burst traces at TR1 and L_{Th}=10 KB.

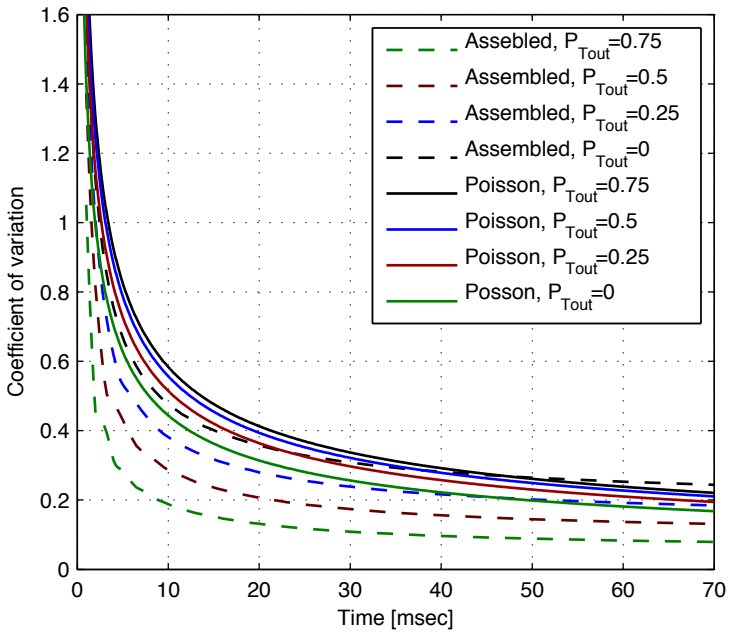

Figure 5.22: Coefficient of variation of number of burst departures as a function of time scale at TR1 and L_{Th}=10 KB.

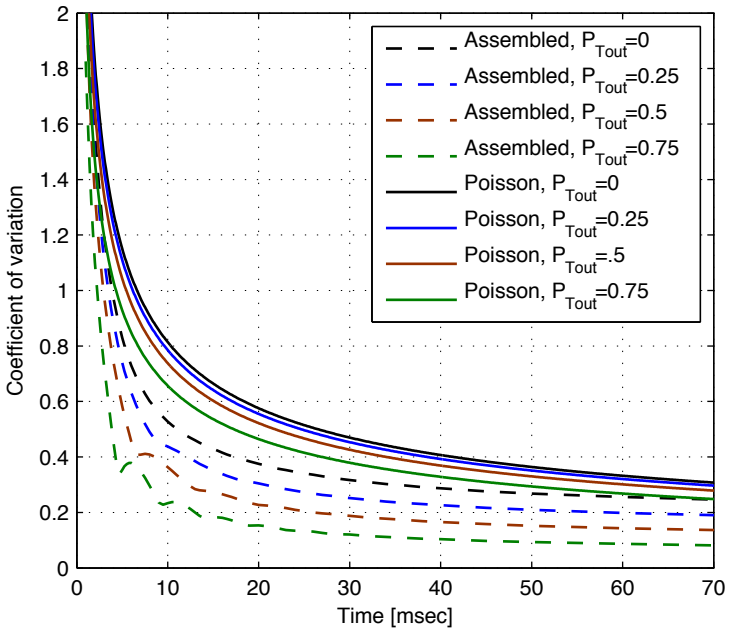

Figure 5.23: Coefficient of variation of number of burst departures as a function of time scale at TR1 and L_{Th}=20 KB.

traffic is affected by the assembler.

Furthermore, the results presented in Figs. 5.22 and 5.23 confirm the analysis carried out on the smoothing impact of burst assembly carried out with simple traffic models in Chapter 4 (compare Figs. 5.22(5.23) with Figs. 4.8(4.9)). The major difference is that the difference between the CoV values of assembled flow and corresponding Poisson flow with the real traces is not as large as that with simple traffic models assumed in Chapter 4.

5.4 Performance Analysis and Discussion

In this section we discuss the results presented in the last section with respect to their importance in design and performance analysis of the OBS network.

First and foremost, we observe that the output of the assembler in this chapter and that presented in Chapter 4 share some similar statistical characteristics. This is, in fact, a confirmation that the conclusions made in Chapter 4 are still valid, in spite of the fact that for the analytical modeling in Chapter 4, we used very simple models – i.e., Poisson – for the input packet-level traffic. More specifically, it is revealed that the assembly process smooths out the burst-level traffic at the short time scales – in the range of the burst assembly time – so that it becomes even smoother than the Poisson process. Note that as demonstrated in Chapter 4, the smoothing effect of the assembly

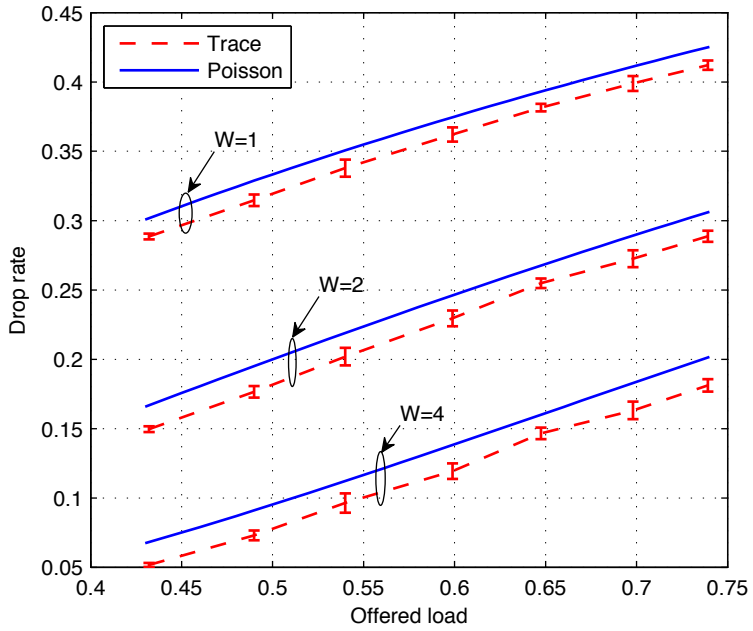

Figure 5.24: Lossrate over the core link with no FDL buffer.

process in short time scales can have significant impacts on the design and performance optimization of OBS networks, which contain either no or limited buffering capacity in forms of optical buffers. In the following we carry out a performance study to show that the same holds true under realistic traffic conditions.

5.4.1 System Model

Our performance analysis experiment is carried out on a single output port of a core OBS switching node having W wavelength channels with the total capacity of D_r Mb/s. The switch supports full wavelength conversion capability and as for the optical buffering, we consider two different scenarios. In the first scenario, the switch does not have any buffering capacity, whereas in the second one the output port of the switch is equipped with a single WDM fiber delay line (FDL) buffer, which can delay a data burst for d *sec* over any of its wavelength channels (see Chapter 3 for the detailed description of the system model). We evaluate the burst loss rate of the considered link while loading it with the assembled burst traffic and compare the results with the case when the burst arrival process is modeled by Poisson. Since in this experiment we are only interested in the impact of the smoothing effect of the assembly on the burst arrival process, the burst sizes in case of Poisson burst arrivals are taken from the assembled burst traces. That is, the difference between the two cases is only in the burst arrival processes.

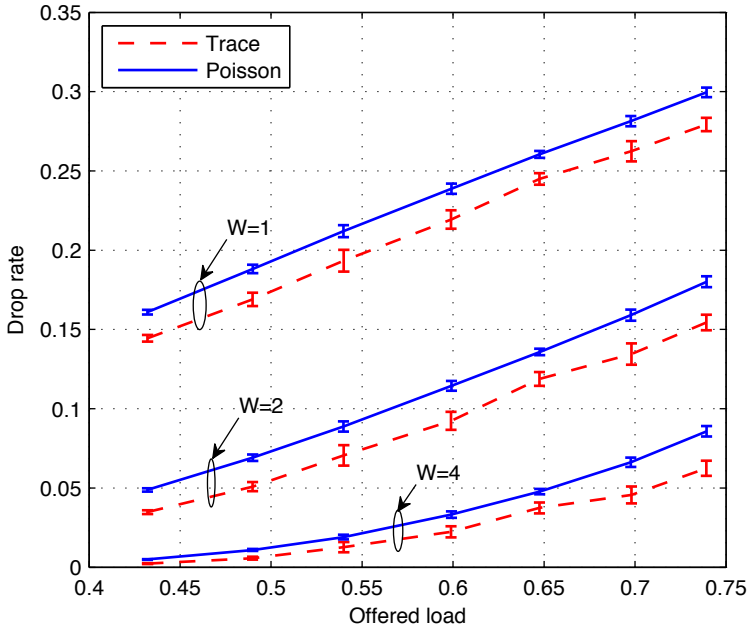

Figure 5.25: Lossrate over the core link with single FDL buffer.

To feed the considered link with the assembled burst traffic, we assume that the link multiplexes 10 traffic flows, each being a burst-level flow resulting from aggregating one of the three packet flows listed in Table 5.1. By varying the number of aggregated TR1, TR2 and TR3 flows – from 0 to 10 – present in the 10-flows combination, we achieve different utilization levels at the output link. To aggregate each packet trace into a burst trace, we select $L_{Th} = 10$ KB and randomly set P_{Tout} in the range $[0, 0.9]$. Further, the resulting burst trace is circularly shifted in time so as to avoid possible synchronization among the burst flows generated from the same packet trace.

5.4.2 Simulation Experiments

For this experiment we develop a simulation model in the OMNeT++. In our simulation we assume $D_r = 300$ Mb/s and d is set to be equal to the transmission time of an average-size burst over the link. Depicted in Figs. 5.24 and 5.25 are the data loss rates as function of the link utilization – with the associated 90 % confidence intervals – at different number of wavelength channels $W = \{1, 2, 4\}$ and for both buffering scenarios. The results for the assembled burst traffic is denoted as $Trace$ and those associated with the Poisson burst arrival is denoted as $Poisson$.

We observe that the loss rate for the case with the assembled burst trace is always smaller than that with the Poisson burst arrival process for the considered scenarios. The difference is noticeable for the link with and without FDL buffers, though the

difference in the former case is larger. Specifically, for the link with single buffer we observe up to more than 100% increase in the loss rate when the trace-based burst arrival is replaced with the Poisson process.

These results are again in agreement with the conclusions arrived at in Chapter 4 regarding the positive impact of burst assembly on the burst drop rate. That is, the burst drop rate on an OBS link under superposition of several assembled burst flows is smaller than that under a Poisson traffic flow, even if the packet traffic flows at the input of the corresponding burst assemblers are correlated.

Part III

Traffic Shaping for Proactive Contention Control

Chapter 6

Virtual Optical Bus

In the previous chapters we demonstrated that in a transport network based on OBS, traffic shaping at the entrance edge of the network offers high potential for contention mitigation. More specifically, the losses due to lack of buffers in transit switches are reduced by buffering – shaping – of data packets at the ingress edge of the network, where we can still use inexpensive electrical buffers. In this part of the thesis, we aim at even more efficient contention reduction by suggesting a novel architecture and protocol for the optical transport network. In our approach, several ingress nodes inside the network are grouped together into a cluster. The set of links connecting the nodes in a cluster form an optical path that is termed *virtual optical bus* (VOB). A path formed in this way is called a virtual bus since the physical links to which a path is associated can be used by several paths simultaneously. A form of coordination among nodes belonging to the same VOB on injecting their packets into the network is applied to eliminate the potential of collisions among packets within the same VOB. In a VOB network – a network that applies the VOB approach – all flows are grouped into VOBs in such a way that each flow belongs to only one VOB and the flows belonging to different VOBs have the minimum potential for contention with each other inside the network.

6.1 System Architecture and Outline of Solution

Consider an OBS transport network (see Chapter3) having N nodes and L unidirectional WDM links, where links connect nodes following an arbitrary topology. Any given link l can support W_l wavelength channels, each operating at rate C Gb/s.

The network operates in the asynchronous burst switching mode. Also, the network employs the store-and-forward switching mechanism, where at each intermediate node data bursts go through a tiny optical delay line while the header packet is converted to the electrical domain, processed by the node controller and converted again back to optics (O/E/O conversion), see Fig. 6.1.

Each node n in the network is equipped with T_n tunable optical transmitters and

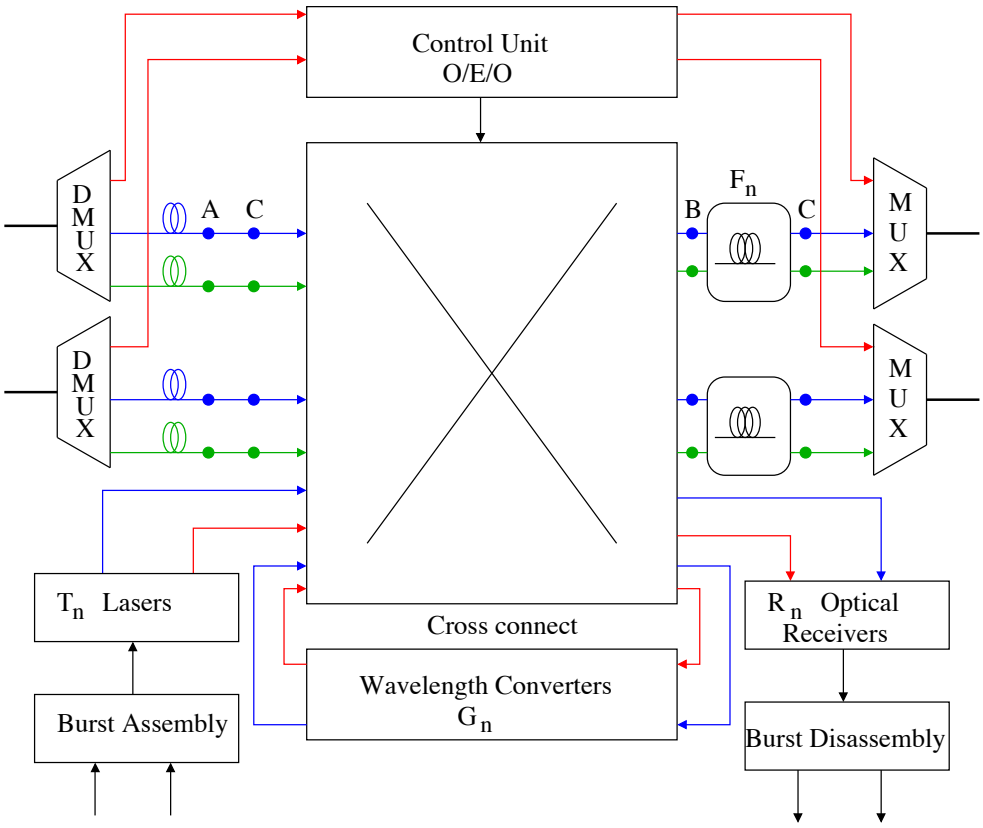

Figure 6.1: An abstract model of the switch assumed in the VOB architecture.

R_n tunable optical receivers. Each tunable transmitter and receiver could be tuned to, respectively, send or receive data on any of the wavelength channels of the links attached to the node. Furthermore, a reactive contention resolution mechanism is implemented in each node n, which incorporates a set of G_n internal wavelength converters as well as F_n WDM fiber delay lines (FDL) per output link of that node. Note that, if a node operates only as a transit node, it does not need to contain any add/drop facilities. Similarly, if a node is only an edge node it does not need any wavelength converter or FDL buffer for contention resolution. Fig. 6.1 shows an abstract model of the node that can operate alternatively as an ingress/egress edge node and as a transit node.

Let us assume temporarily that the ingress nodes work independently – with no coordination among each other – for injecting traffic into the network, which is natural for packet switching paradigm. This carries a natural potential of collision among packets in the network. A collision occurs when two or more data packets are going to use the same channel on a given link for an overlapping period of time.

In electronic packet-switched networks, efficient avoidance of this adverse phenomenon is usually done by introducing some buffers that hold colliding packets during

times of contentions. Unfortunately, the all-optical nodes have limited buffering potential – through FDL buffers – which due to lack of random access property can offer only pre-defined fixed delay values. Accordingly, the only way to limit the contention is an appropriate traffic shaping at the input edge of the network. The potential of this approach has been demonstrated in the previous chapters. An optimal idealistic approach for this purpose would be global scheduling of all sources so that every single packet transmission from any source in the network is coordinated with all other sources so as to avoid collision in the transit nodes. This could – theoretically – be realized by introducing a central controller to the network that schedules all packet transmissions in the network in a collision-free manner. In this way, up to some maximum utilization the controller can ensure that the packet will not collide with any other packet along its path to destination. Although this approach would potentially lead to the highest throughput in the network and the minimum amount of collisions, this might be too costly and not practical due to the necessary ideal time synchronization of all components, complexity as well as signaling and scheduling overhead. A natural alternative is decomposition of the global scheduling of all sources into smaller sets with local scheduling within each set and hoping that the results of the local scheduling will not collide too much with each other. On account of this, we postulate grouping of traffic sources into disjoint clusters and apply a form of coordination – traffic shaping – among associated sources in each cluster. In this way, only the groups of sources remain independent and subject to collisions, thus reducing the collision rate in the network.

A crucial issue for clustering is setting the requirements for assigning sources to clusters. Taking into account that the final goal is to decrease the packet collisions in the network, a natural requirement is to form a cluster out of the sources that potentially share the same set of links in their path and thus their packets are likely to collide with each other. Furthermore, we have to take care that the sets of routes of flows from different clusters should intersect minimally – ideally be disjoint. In the latter case there would be – by definition – no collisions between the sources belonging to different clusters either! This might, however, be impossible in reality, thus we suggest keeping the inter-cluster collisions on a low level by finding paths for them with minimal interaction.

Concerning the scheduling within the clusters, we need a local scheduling mechanism that is feasible and at the same time efficient. In fact, as opposed to the global scheduling, the local scheduling pertains only to a limited number of flows in the network and has therefore much lower complexity. Nevertheless, we will go a step further and target a distributed local scheduling mechanism rather than a centralized approach. As for the efficiency, the same criteria are applied to the local scheduling as the global scheduling. That is, a local scheduling algorithm operating on a local set of sources should schedule the corresponding packets with minimum – ideally no – collisions within that local set.

To evaluate the effectiveness of our approach in improving the performance of the

network we consider three metrics: packet loss rate, access delay of packets and network throughput. Packet loss rate will be calculated as the fraction of generated packets by the ingress nodes that are lost in the network due to collisions. The access delay is calculated as the difference between the time a packet is generated by an ingress node and the time it is released to the network, and is calculated for successfully delivered packets. The maximum achievable throughput of the network will also be evaluated and compared with that of a classical OBS network.

6.2 Virtual Optical Bus Description

Our approach for coordination of flows within each cluster will be based on the abstraction of the virtual optical bus (VOB) – a virtual unidirectional bus accommodating a set of origin-destination (O-D) flows, defined as a flow of packets originated at an ingress edge node destined to a specific egress node in the considered network, on their entire route.

The following features collectively give the formal definition of the VOB.

- A VOB is defined over a specified sequence of connected links, i.e., it is a directed simple path with a single origin and destination.
- Each VOB is piecewise associated with a specific wavelength channel. On a multichannel link with wavelength converters available, a VOB can use any channel on that link; however, flows associated with that VOB may not use more than one channel at any given time.
- Each O-D flow must be associated with one and only one VOB.

Note that the third feature implies that both origin and destination of a given flow should be associated to the same VOB, meaning that traffic of an O-D flow will not be split/switched over multiple VOBs.

Following the description given above, establishing the coordination among flows associated to a VOB is now translated into designing a medium access control (MAC) protocol that ensures a collision-free, efficient and fair access to the VOB for the associated traffic flows. For this purpose we use a MAC protocol that is based on buffer-insertion protocol, which was first introduced in mid 80s and was afterwards improved and used in many other works, see [164]. In the buffer-insertion, each intermediate active node on a bus – an intermediate node that injects traffic into the bus – is equipped with an extra buffer called insertion buffer. The buffer is used to actively delay transit packets flowing on the bus in order to avoid them from colliding with local packets in transmission. Also, at every intermediate node priority is given to the transit traffic over the local traffic. In this way, the required capacity of the insertion buffer would be small. Specifically, it must hold a transit packet until the transmission of an ongoing

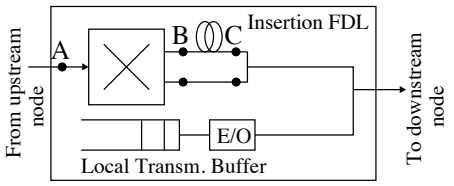

Figure 6.2: An abstract model of a single ingress node equipped with a FDL insertion buffer on the transit path. There are two possibilities for incorporating the insertion buffers into the architecture of the switch depicted in Fig. 6.1: at the input or output of the switch (see points marked as A, B, and C). In the former case, the insertion buffers are placed between the delay lines and the switch fabric, which necessitates additional 1×2 switching elements before the insertion buffers to allow for bypassing the insertion buffers if required. In the latter case, the insertion buffers could be incorporated into the FDL buffering unit. In either case, additional ports on the main switch would be needed.

local packet is completed; therefore, it only needs the capacity equal to the transmission time of a maximum-length packet.

A known phenomenon associated with the buffer insertion protocol is the fairness issue. In fact, in a physical bus using the buffer insertion MAC protocol, those nodes closer to the beginning point of the bus have better and faster access to the bus [165]. This might result in an unfair access to the bus for the nodes closer to the end of the bus. Nevertheless, this is not the case for the VOB. In fact, as discussed in the next section, in designing a VOB network O-D flows are associated with a VOB based on their average data rate; that is, it is assumed that a single upstream node does not have the potential to monopolize the VOB in long term. Nevertheless, in order to guarantee that downstream O-D flows do not have to wait excessively long times before they are granted access to the VOB, in our design each source node in the network is further equipped with a token bucket shaper.

In order to adopt the buffer-insertion protocol in VOB, we need an insertion buffer per VOB per any active intermediate node being part of that VOB. That is, in a network with N nodes each node would require at most $N-1$ insertion buffer since it generates traffic destined to at most $N-1$ other nodes in the network. Since the insertion-buffer is used to buffer transit optical packets already traveling over the bus, it must be realized in the optical domain to avoid the need for the electro-optical conversion. Taking into account that the required capacity of an insertion buffer is small, it can be easily realized by single FDL as shown in Fig. 6.2.

The operation of the VOB MAC protocol is as follows.

- At any node i being part of the VOB there is a token bucket shaper that is filled with new tokens at rate t_i (B/s). The bucket has the maximum size of $T_{max,i}$ (B) and T_i (B) denotes the available tokens in the bucket at any given time.

- Any active node i on the VOB can inject a packet, which is at the head of its local transmit queue, into the VOB as soon as the following three conditions are met: *i*) the corresponding transmission channel is idle, *ii*) the packet size is smaller than the available token T_i in the corresponding bucket and *iii*) the insertion FDL is idle or the gap between the time that the packet inside FDL starts leaving the buffer and the current time is larger than the time required to transmit the local packet. If the active node is the most upstream node of the VOB, it does not have an insertion buffer and therefore it does not need to consider the condition iii. In any case, immediately after the packet transmission is scheduled, T_i is reduced by the packet size.

- If a transit packet arrives at an intermediate node of the VOB while a local packet is being transmitted, it will be delayed by the insertion FDL.

This operation guarantees a collision-free packet transmission within a VOB. In the next chapter, we investigate the operation and performance of the proposed MAC protocol under different settings in detail.

To elaborate more about the VOB operation let us now make an example. Fig. 6.3 depicts a network with 5 nodes, in which nodes 1–4 generate traffic destined to node 5 each with the intensity equal to 35% of the capacity of a single WDM channel. Packets generated at each node go through an electro-optical (E/O) conversion before transmission. The link $(4 \rightarrow 5)$ is offered the load of 1.4 through multiplexing four independent O-D flows. If we suppose that this link has two data channels and two tunable wavelength converters with no FDL buffer, and that traffic of each source is injected according to the Poisson process, by applying the known Erlang-B formulae [105] we will observe some 29% packet loss rate on this link under OBS operation. Now, we establish two VOBs over the physical routes $(1 \rightarrow 2 \rightarrow 4 \rightarrow 5)$ and $(3 \rightarrow 4 \rightarrow 5)$, respectively (Fig. 6.3-b). The former VOB accommodates O-D flows $(1 \rightarrow 5$ and $2 \rightarrow 5)$ and the latter one carries the traffic of O-D flows $(3 \rightarrow 5$ and $4 \rightarrow 5)$. In this case, by using the MAC protocol described above node 2 (4) will schedule its local packets in a way that they do not overlap with packets of node 1 (3), i.e., no intra-VOB collision occurs. Consequently, we only have two independent traffic flows being multiplexed into the link $(4 \rightarrow 5)$ and therefore no packet collision will occur on this link anymore, since it already has two channels.

6.3 VOB Network Design

Design of a VOB network consists of grouping all O-D flows into VOBs, which has to be done with the objective of minimizing the packet collision rate in the network. We refer to this step as VOB layout design, which can formally be stated as follows. Having a network with given resources, as explained in Section 6.1, and an O-D traffic matrix, how to cover the traffic matrix by a set of VOBs, with respect to the limitations

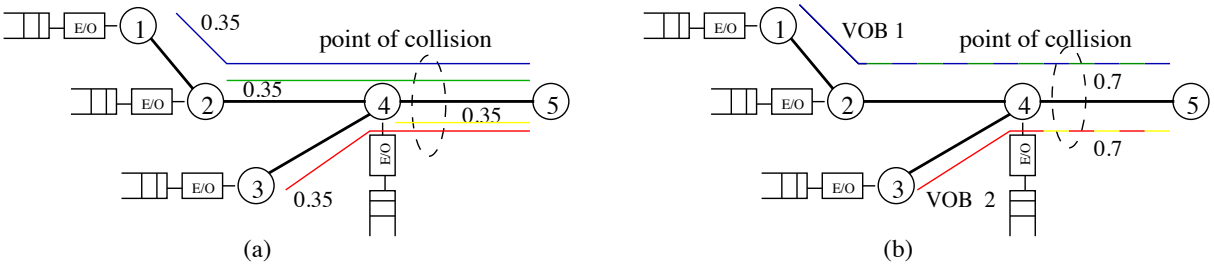

Figure 6.3: An example of applying the VOB Framework, (a) a simple network with 4 O-D flows, (b) the same network after applying the VOB framework.

imposed by the topology and the resources available within each node and link, in such a way that the packet collision in the network is minimized. The VOB layout design should give us complete information about the beginning, the end and the route of all VOBs as well as the IDs of flows to be covered by each VOB. It should be noted that VOB layout design involves assigning O-D flows to VOBs and that, in turn, implies routing of O-D flows in the network. In general, there are two possibilities for routing bidirectional flows between two specific nodes in the network: symmetric or asymmetric routing. In our design, we do not enforce symmetric routing and this can cause some bidirectional flows to take different routes on opposite directions. Nevertheless, for some scenarios such as the ring network topology the symmetry between the routes are to a high extent achieved.

Taking into account that the intra-cluster collisions – collisions among the flows belonging to the same VOB – are completely suppressed by the proposed MAC protocol, the objective function of the VOB layout design translates into minimizing the inter-cluster collisions across the network. Nevertheless, minimization of the inter-cluster collisions cannot be solved directly and in a straightforward manner, if at all possible, since it depends on many factors such as traffic characteristics and routing. Consequently, we resort to an alternative approach, where the objective function is set to minimization of maximum number of VOBs that are multiplexed to any link in the VOB network. We recall from Section 4.5 that the drop rate on a link is in direct relation with the number of flows being multiplexed onto that link. In fact, this objective function will minimize the interaction among routes of VOBs in the network, thereby reducing the inter-cluster collision across the network.

In the following, we apply the path-based method [166] to formulate the VOB layout design as an integer linear programming (ILP) problem. In order to facilitate finding a solution to the problem, we try to limit the optimization search space. This is achieved through generating a list of many potential VOBs in the network and also for each potential VOB determining all O-D flows that it could support. This process is carried out in a preprocessing phase as explained below.

6.3.1 Preprocessing

Let us assume that all links and O-D flows in the network are, respectively, populated in set $\mathbb{L}$ with L elements, and $\mathbb{F}$ with F elements. We first apply the k-shortest path (KSP) routing algorithm [167] to the given topology and populate the set of all k-shortest paths between all pairs of nodes in the network in $\mathbb{P}$ with size P. Note that $\mathbb{P}$ is the set of VOB candidates in the network.

Now we form the matrix Θ $(P \times L)$ that relates links in the network to the paths populated in $\mathbb{P}$. That is,

$$\theta(p,l) := \begin{cases} 1 & \text{if link } l \in \mathbb{P} \text{ is used in path } p \in \mathbb{P}, \\ 0 & \text{otherwise.} \end{cases}$$

In the next step, we form Γ, which is a $P \times L \times F$ matrix, and whose elements are calculated as:

$$\gamma(p,l,f) := \begin{cases} 1 & \text{if O-D flow } f \in \mathbb{F} \text{ could be supported} \\ & \text{by path } p \in \mathbb{P} \text{ on link } l \in \mathbb{L}, \\ 0 & \text{otherwise.} \end{cases}$$

In case there are any constraints regarding the routing of some O-D flows in the network they have to applied during the formation of Γ. One such constraint can be the maximum allowable hop count for routing of O-D flows. We finally note that for a given network topology the preprocessing phase needs to be done only once.

6.3.2 ILP Formulation

Assumption

For our formulation we make the assumption that all nodes support full wavelength conversion. Note that, this is not mandatory for the VOB design framework, and we only make the assumption here since it will improve the performance and further simplify the design of the layout

Parameters and Variables

The parameters used in the formulation are the following:

- $\Lambda = [\lambda^f]$ is the given demand matrix, where λ^f denotes the intensity of the O-D flow $f \in \mathbb{F}$ (normalized to the wavelength capacity). It is assumed that $\lambda^f < 1$ for $f \in \mathbb{F}$.
- $A_{max,w}$ is the maximum allowable load on a single WDM channel in the network.

Also, the variables are:

- x_p

$$x_p := \begin{cases} 1 & \text{if path p} \in \mathbb{P} \text{ is a VOB,} \\ 0 & \text{otherwise.} \end{cases}$$

- y_p^f

$$y_p^f := \begin{cases} 1 & \text{if flow f} \in \mathbb{F} \text{ is supported by path p} \in \mathbb{P}\ , \\ 0 & \text{otherwise.} \end{cases}$$

Objective Function

We set the ILP objective function to

$$Min\ \max_{l \in \mathbb{L}} \sum_{p \in \mathbb{P}} \theta(p,l) x_p. \tag{6.1}$$

The objective function minimizes the maximum number of multiplexing VOBs over any link in the network.

Optimization Constraints

$$\sum_{p \in \mathbb{P}} y_p^f = 1 \qquad \forall f \in \mathbb{F} \tag{6.2}$$

$$\sum_{f \in \mathbb{F}} y_p^f \lambda^f \gamma(p,l,f) \leq x_p A_{max} \qquad \forall l \in \mathbb{L}, p \in \mathbb{P} \tag{6.3}$$

In constraint 6.2 each traffic flow is associated to one and only one VOB. Also, constraint 6.3 ensures that traffic offered to any part of a VOB on any link in the network does not exceed a fraction of WDM channel capacity equal to A_{max}, which is a design parameter used to control the access delay of traffic sources. Note that A_{max} should be set to a value larger than the $\max_{f \in \mathbb{F}} \lambda^f$. As we will discuss in the next chapter, this parameter plays in important role in the performance of the designed VOB networks.

6.4 Related Work

Although there is a large number of publications on contention resolution issue of OBS networks, only a few of them consider the proactive measures to reduce packet collisions rate. In this section we outline the most relevant approaches and compare them with VOB.

The authors in [168] present a new packet-based optical transport architecture based on light trail (LT), which is a generalized form of the light path (optical circuit) where intermediate nodes along the path of the circuit may also inject traffic into it. Therefore, a LT could also be considered as an optical bus that facilitates traffic

grooming in the network. Our approach is essentially different from the LT. Specifically, in the LT approach the whole capacity of a wavelength channel must be dedicated to each LT on any link in the network being part of that LT and there is no switching in the intermediate nodes. That is, the LT approach allows only for intra-wavelength multiplexing. With the VOB approach, however, a single wavelength on a link may be shared by all the VOBs on that link, i.e., both intra and inter-wavelength multiplexing are realized. On the other hand, in order to transmit any packet using LT one needs to establish a LT in advance, whereas in the VOB, establishing a virtual bus can take place in the background while transmission of packets are in progress.

In [169] the authors present a proactive burst scheduling algorithm aiming at reducing the collision rate in the network. In this approach, every node injects its local packets into the network in a way that they do not overlap in time with each other if the packets are supposed to traverse the same path. Our approach based on VOB differs from proactive burst scheduling, in that the latter one only does a local coordination at every ingress node through implementing a new MAC protocol, assuming that routings of the O-D flows are already fixed. In the VOB-based approach, however, a network-wide traffic engineering is applied, which also involves finding appropriate paths for the flows to reduce the potential of collisions in the network.

Dual bus optical ring (DBORN) is an architecture for optical metro transport networks that has been studied in [170]. A metro network based on DBORN is composed of two WDM rings, where one is used as the working ring and the other one as the back up. WDM channels on each link are divided into upstream and downstream channels. All nodes attached to the ring are passive, with the exception of one node known as a hub, being the only node which can transmit traffic over downstream channels and receive or remove traffic over upstream channels. All of the passive nodes have access to the upstream channels using an optical carrier sense multiple access with collision avoidance (CSMA/CA) MAC protocol. In order that node i transmit a data packet to node j, where i and j are both attached to the ring, node i has to first transmit the packet to the hub over one of the upstream channels and then the hub node would forward the packet to node j on one of the downstream nodes. In DBORN, both the transmission and the packet processing of the whole network are limited by those of the hub node. Also, the resources in the network are not utilized efficiently, since for example to transmit a packet from a given node to its adjacent node, the packet has to travel the whole circumference of the ring network first. In contrast to DBORN, VOB can achieve a high network utilization since source-destination pairs communicate directly without using a hub. In addition, DBORN is a ring-based approach and is not extended to an arbitrary meshed topology, whereas VOB architecture can be applied to any meshed topology.

The authors in [171] present time-domain wavelength interleaved networking (TWIN) as a new optical transport architecture. TWIN consists of smart edge nodes that are equipped with ultra-fast tunable lasers and passive fixed-routing optical core nodes. A unique wavelength is assigned to each destination node in the network, which by the

way, serves as the address of that node. In order to transmit a packet from node i to node j, the node i has to tune its tunable laser to the wavelength associated to node j for the duration of the packet transmission. All the passive intermediate nodes are configured in advance to guide incoming packets towards their destinations based on the wavelength channel that the packets are arriving on. In this way, a multipoint-to-point tree is formed to every egress node in the network. The key design issue of the TWIN architecture is to devise a scheduler that can handle a collision-free transmission of packets on a given destination tree. Designing an appropriate scheduling algorithm that can work efficiently under highly dynamic traffic is a challenging issue. In fact a fixed scheduling for all nodes of a tree has to be calculated before any packet transmission can take place on that tree. This is, however, not required in VOB architecture,which instead of trees uses busses, which are much easier to coordinate. Additionally, the number of wavelength channels in TWIN must always be at least equal to the number of nodes in the network, whereas in VOB there is no constraint on the required number of wavelengths in the network.

Chapter 7

Performance Analysis of VOB

In this Chapter, we evaluate the VOB MAC protocol and compare it to that of classical OBS. The VOB MAC protocol presented in Chapter 6 is a new variation of the well-studied buffer-insertion protocol. In the buffer-insertion, where a physical bus is shared among a set of nodes, 100% throughput can be achieved as long as the total load offered to each segment of the shared bus is smaller than transmission capacity of that segment. The only concern would be delay of accessing the bus, which rapidly increases with the distance of a node from the head-end of the shared bus. This is the so-called fairness issue associated with the buffer-insertion protocol. The issue stems basically from the inherent priority of upstream sources in accessing the bus that allows them to grab as much capacity as they want without considering the needs of downstream nodes.

This can effectively be controlled by limiting the injection rate of each source to the bus, e.g., by means of a token bucket controller. Nevertheless in a VOB with load-controlled insertion-buffer protocol, 100% throughput cannot be achieved, since FDL-based insertion-buffer cannot utilize the whole capacity of an optical link. That the type of buffer used in realizing the FDL-based insertion buffer does not support random access property leads to the fragmentation of the data channel and this consequently hinders the optimum utilization of channel capacity. Therefore, it is of great importance to investigate the delay-throughput characteristics of the MAC protocol proposed for the VOB.

7.1 System Model

Let us consider the delay-throughput of a single VOB on a link in the network. The performance of the VOB MAC protocol on this link depends on several factors, including the number of VOBs multiplexed onto the link and their associated traffic intensity, the number of wavelength channels on the link and the number of upstream nodes. Therefore, we need a sophisticated scenario that allows us to consider all these parameters. For this purpose we have designed a test scenario as depicted in Fig. 7.1.

The network in Fig. 7.1 is a single-bottleneck topology that consists of B parallel

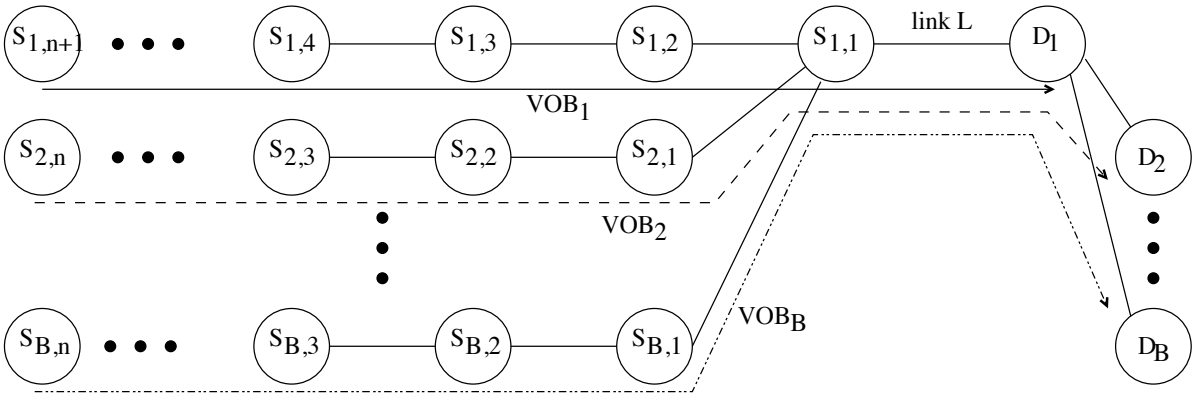

Figure 7.1: Single-bottleneck scenario used for MAC protocol evaluation.

branches of nodes, $(n+1) \times B + 1$ nodes and $(n+1) \times B$ unidirectional links, where a link connecting any two nodes is realized by a WDM link with W_l data channels and one control channel. We further make the following assumptions:

- Nodes $S_{i,1}, ..., S_{i,n+1}$ generate traffic destined for node D_i $(1 \leq i \leq B)$.
- Aggregate load generated for node D_i $(1 \leq i \leq B)$ is less than the transmission capacity of a single wavelength channel.
- The network operates under either of the two modes: classical OBS and VOB. In the latter case, one VOB is established on any branch i $(1 \leq i \leq B)$ and all nodes on the branch are associated to that VOB, see Fig. 7.1 . Accordingly in this case each active node on the VOB is assumed to be equipped with the VOB MAC protocol and corresponding FDL-based insertion buffer. In addition, it is assumed that the local transmission buffer at each node has an unlimited buffering capacity.
- The number of wavelength channels on any link at the left side of the bottleneck link L is equal to n, and that of the bottleneck link L is equal to W_L. Each link on the right side of link L has 1 wavelength channel.
- Each source node is equipped with one tunable optical transmitter, n tunable wavelength converter and it does not have any FDL buffer for contention resolution.
- At each source node, bursts of fixed length are generated according to the Poisson process with intensity ρ (the same for all sources), as normalized to wavelength channel capacity. At the end of this chapter we will study the impact of burst assembly on the VOB performance.

7.2 Simulation Experiments and Discussion

To quantify the performance of the above scenario, we developed a simulation platform of the network using the discrete event simulator OMNeT++. Simulation experiments are carried out for both OBS and VOB network operation at a variety of combinations of $B = \{2, 3, 4\}$, $n = \{2, 3, 4, 6\}$, $W = \{1, 2, 3, 4\}$ and at different load values. In the experiments, wavelength channel capacity and burst length are set to 10 Gb/s and 10 KB, respectively. Also, the token arrival rate parameter of the token bucket shaper in each source node is set to be equal to 1.1 times the average load value of that source and the bucket size is set to allow transmission of up to 20 consecutive packets for each source.

We evaluate the packet loss rate, throughput and access delay of the designed scenario. The loss rate is estimated at the entrance of the bottleneck link L, as this is the only point in the network that packets might get dropped due to contentions. The throughput is considered at the network level and the delay is estimated as experienced by node $S_{1,1}$ in accessing VOB_1.

In the analysis given below, we distinguish between two modes of operation for VOB: mode I corresponding to the case where number of VOBs B associated with the bottleneck link L is smaller than or equal to W_L (i.e. $W_L \geq B$) and mode II referring to the other cases, i.e. $W_L < B$.

Results of our experiments are depicted in Figs. 7.2-7.12, which show throughput, access delay and loss rate curves as function of load offered to link L for different combinations of B, n and W.

7.2.1 Loss Rate and Throughput

Figs. 7.2-7.9 depict the loss rate on the bottleneck link and the network throughput as a function of load offered to the bottleneck link. First and foremost, it is observed that applying the VOB greatly reduces the loss rates and as a result, the achieved throughput of the VOB network is higher than that of OBS network.

In fact, in VOB mode I the loss rate is always zero and therefore the throughput is equal to the offered load. This holds true for any number of nodes n and any number of VOBs B. Therefore, in this mode, we observe the largest difference between the VOB and OBS networks in terms of the loss rate and the throughput. In mode II, burst losses occur under both VOB and OBS networks, and the loss rate increases with the offered load. Nevertheless, the loss rate of VOB network is still much smaller than that of OBS network. In addition, in this mode the difference between the two approaches in terms of the throughput increases with load. Table 7.1 summarizes the throughput gain of VOB network over OBS network under different settings of the parameters and at 70% bottleneck-link load. It is observed that the throughput gain of VOB for the considered scenario can be as high as 22.71%.

Now let us consider impacts of number of nodes n. Although in VOB mode I both

Table 7.1: Throughput gain of VOB over OBS at 70% link load in percent.

	B=2		B=3		B=4		
	$W_L = 2$	$W_L = 1$	$W_L = 3$	$W_L = 2$	$W_L = 4$	$W_L = 3$	$W_L = 2$
n=2	16.5	2.9	13.0	7.4	7.0	8.0	4.1
n=3	19.7	5.3	15.6	9.1	9.2	9.4	5.4
n=4	22.2	5.7	16.7	10.0	10.1	10.0	6.1
n=6	22.7	8.8	17.8	10.4	10.9	10.6	6.9

throughput and loss rates are independent from the number of nodes n, this is not the case for OBS, in which both the loss rate and the throughput deteriorate with n. In other words, in this mode the gain of VOB network over OBS network – in terms of throughput and loss rate – increases with n. In VOB mode II, similar to OBS, the throughput of VOB network decreases with n for a given setting. Nevertheless, the reduction of throughput for VOB network is much smaller than that of OBS, which helps the VOB network to outperform OBS more largely with n.

Finally, we analyze the impacts of number of VOBs. From Table 7.1 it is seen that the throughput gain of VOB network over OBS network in mode I decreases with B. For instance, at n=6 the throughput gain reduces from 22.7% at B=2 to 10.9% at B=4. To clarify the reason we note that in this mode the number of VOBs B is equal to the number of wavelength channels on the bottleneck link. That is, at B=4 the number of wavelength channels on the bottleneck link is twice larger than that at B=2, whereas the normalized offered load is fixed. On the other hand, at a fixed load the drop rate should decrease with the number of available wavelength channels. That is, the drop rate of OBS network decreases with B, while the drop rate of VOB network is fixed and equal to zero in this mode. As a result, the difference between the VOB and OBS networks reduces with B. In VOB mode II the throughput gain depends very much on the ratio B/W_L as depicted in the Table 7.1. For instance, the throughput gains are similar for cases $B = 4/W_L = 2$ and $B = 2/W_L = 1$. Additionally, the throughput gain of VOB network improves as B/W_L approaches 1.

7.2.2 Access Delay

In the last section it was demonstrated that VOB network achieves a better performance – in terms of loss rate and throughput – in comparison with the classical OBS. This gain, however, comes at the cost of increase in the VOB access delay, i.e. the time it takes for a source node to inject its traffic into the network. In this section, we investigate the access delay characteristics of VOB.

To characterize the access delay of the VOB at a given node it is important to take into account impacts of number of upstream nodes, which actively inject traffic into

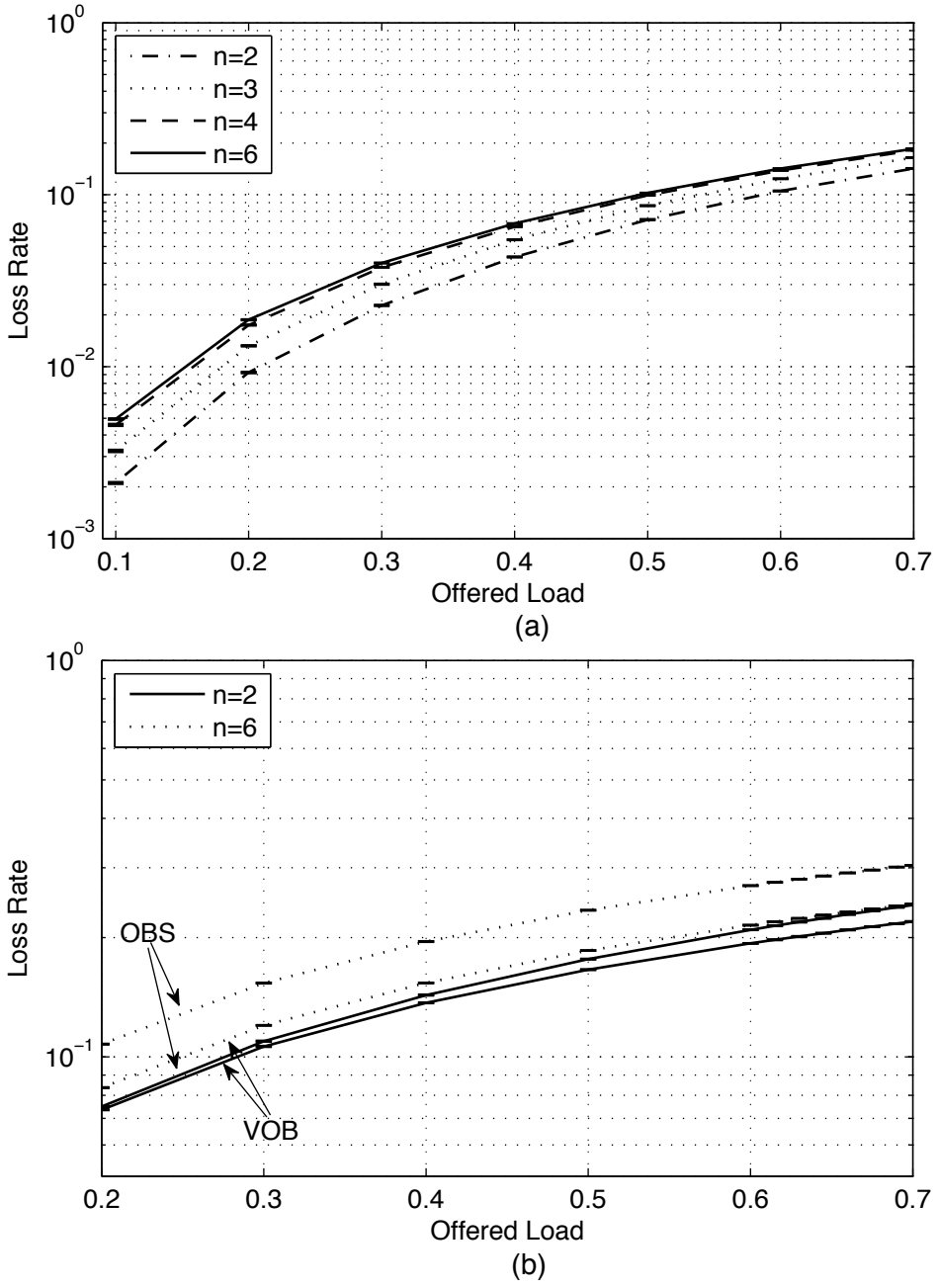

Figure 7.2: Loss rate over the link L against the offered load for B=2, (a) OBS, W=2. The loss rate for VOB is zero, (b) W=1.

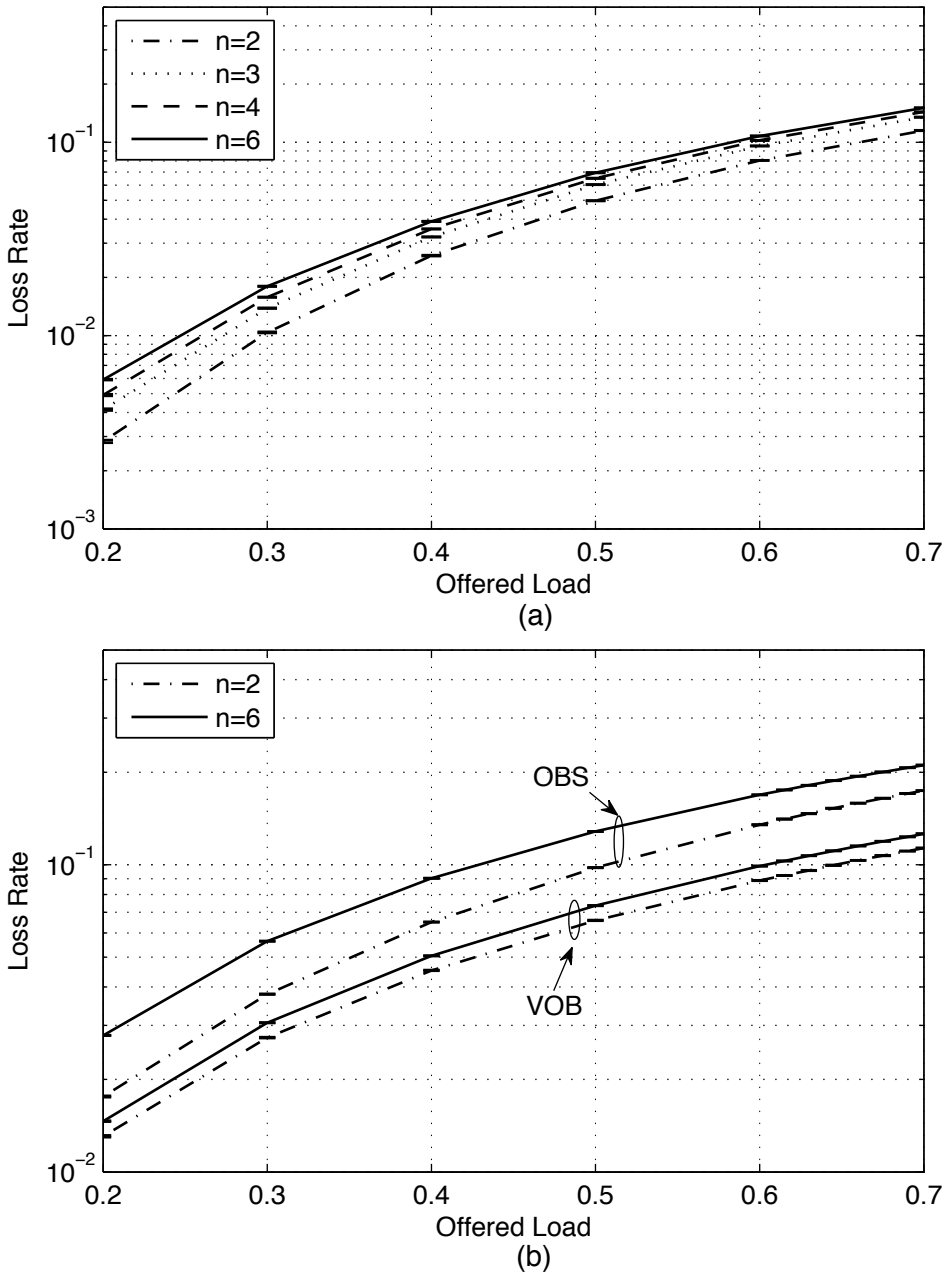

Figure 7.3: Loss rate over the link L against the offered load for B=3, (a) OBS, W=3. The loss rate for VOB is zero, (b) W=2.

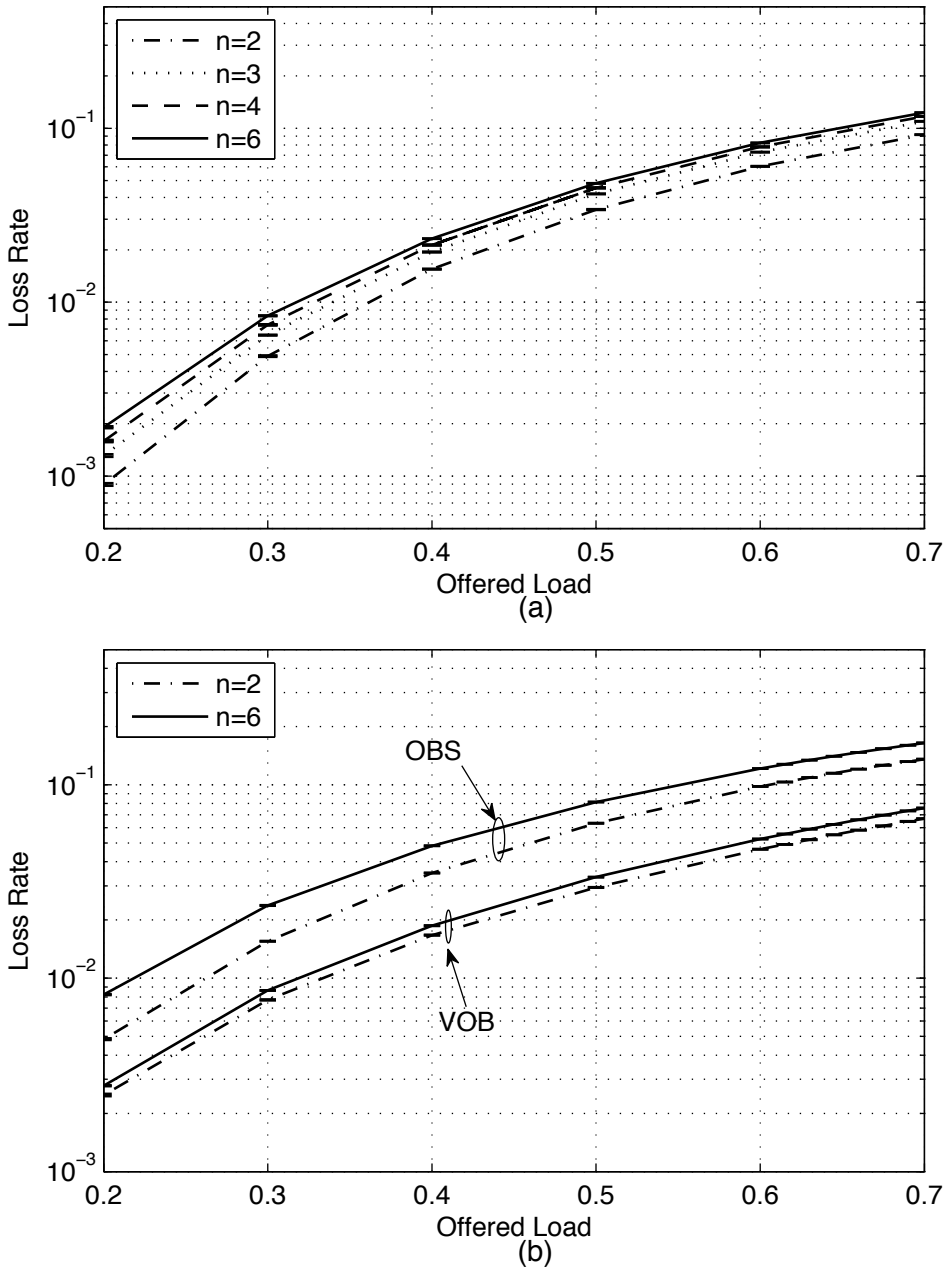

Figure 7.4: Loss rate over the link L against the offered load for B=4, (a) OBS, W=4. The loss rate for VOB is zero, (b) W=3.

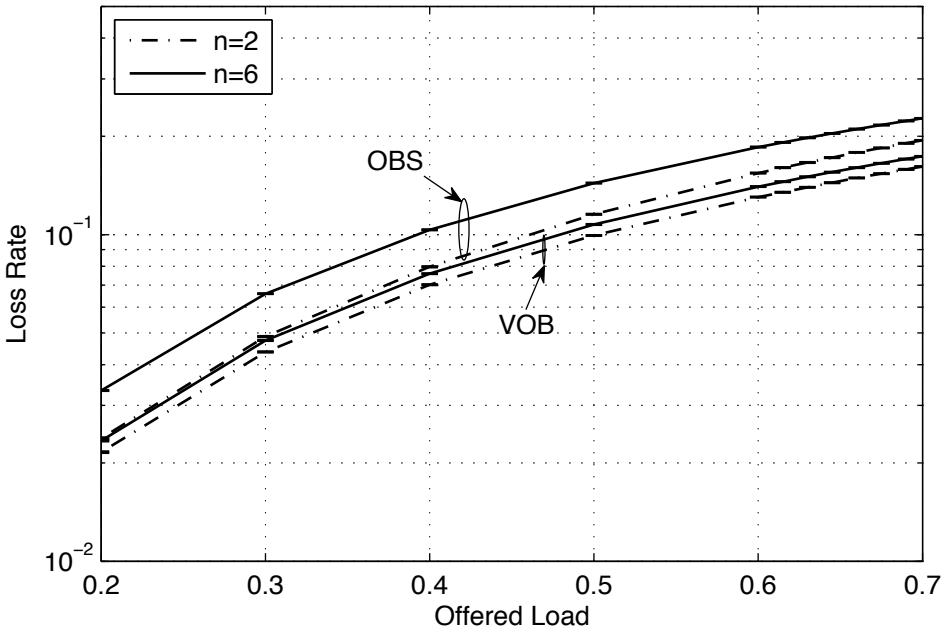

Figure 7.5: Loss rate over the link L against the offered load for B=4 and W=2.

the VOB. In our analysis the access delay is estimated at node $S_{1,1}$ while accessing VOB_1 as a function of total load offered to VOB_1 on the bottleneck link. To account for the different number of upstream nodes, we vary n such that at a given value of $n = N$ there would be N nodes upstream to node $S_{1,1}$.

Figs. 7.10-7.12 depict the average access delay of node $S_{1,1}$ as a function of total load offered to VOB_1 on the bottleneck link for different combinations of n and B in mode I of operation. The access delays for VOB in mode II (i.e. $W_L < B$) as well as for OBS operation are smaller than 10 μs for all settings of the considered scenario and therefore not shown here.

It is seen that the access delay for node $S_{1,1}$ is smaller than 150 μs at 70-75% VOB load even when there are 6 upstream nodes. It should be noted that the access delay in the range of few hundred μs is considered acceptable, taking into account the propagation delay in a metro/core network that is usually in the range of tens of ms.

More specifically, we observe that the access delay increases smoothly with VOB load up to a knee point, which occurs at around 60-65% load. At the load values beyond this point the access delays increase sharply. Although the number of upstream nodes has a large impact on the access delay after the knee point, this impact is not large when the VOB load is smaller than the load value of the knee point. Furthermore, we observe that that number of VOBs B does not affect the access delay to a large extent, which is not surprising, since different VOBs are to a high extent isolated from each

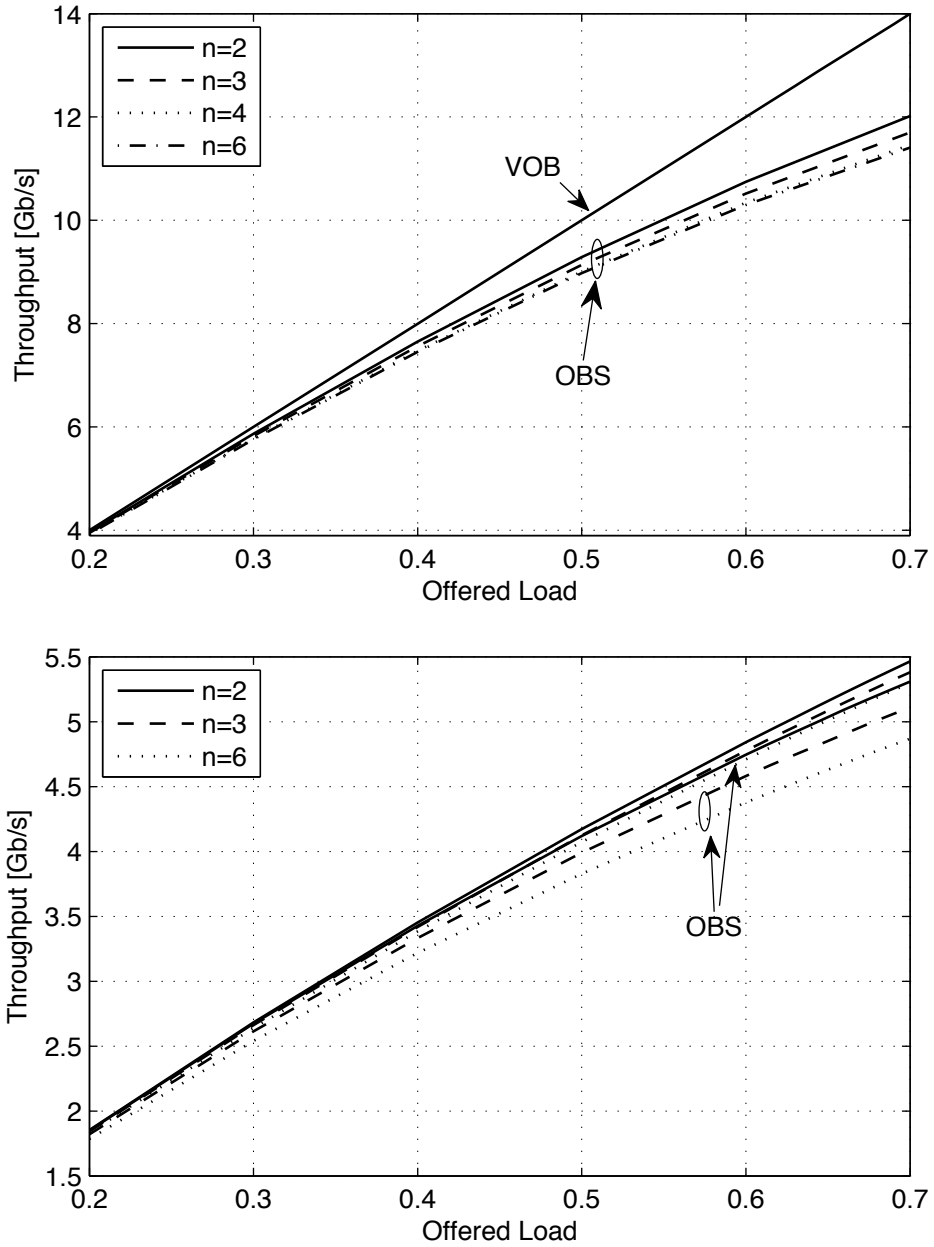

Figure 7.6: Throughput of the network against the offered load to link L for B= 2, (a)W=2, (b) W=1.

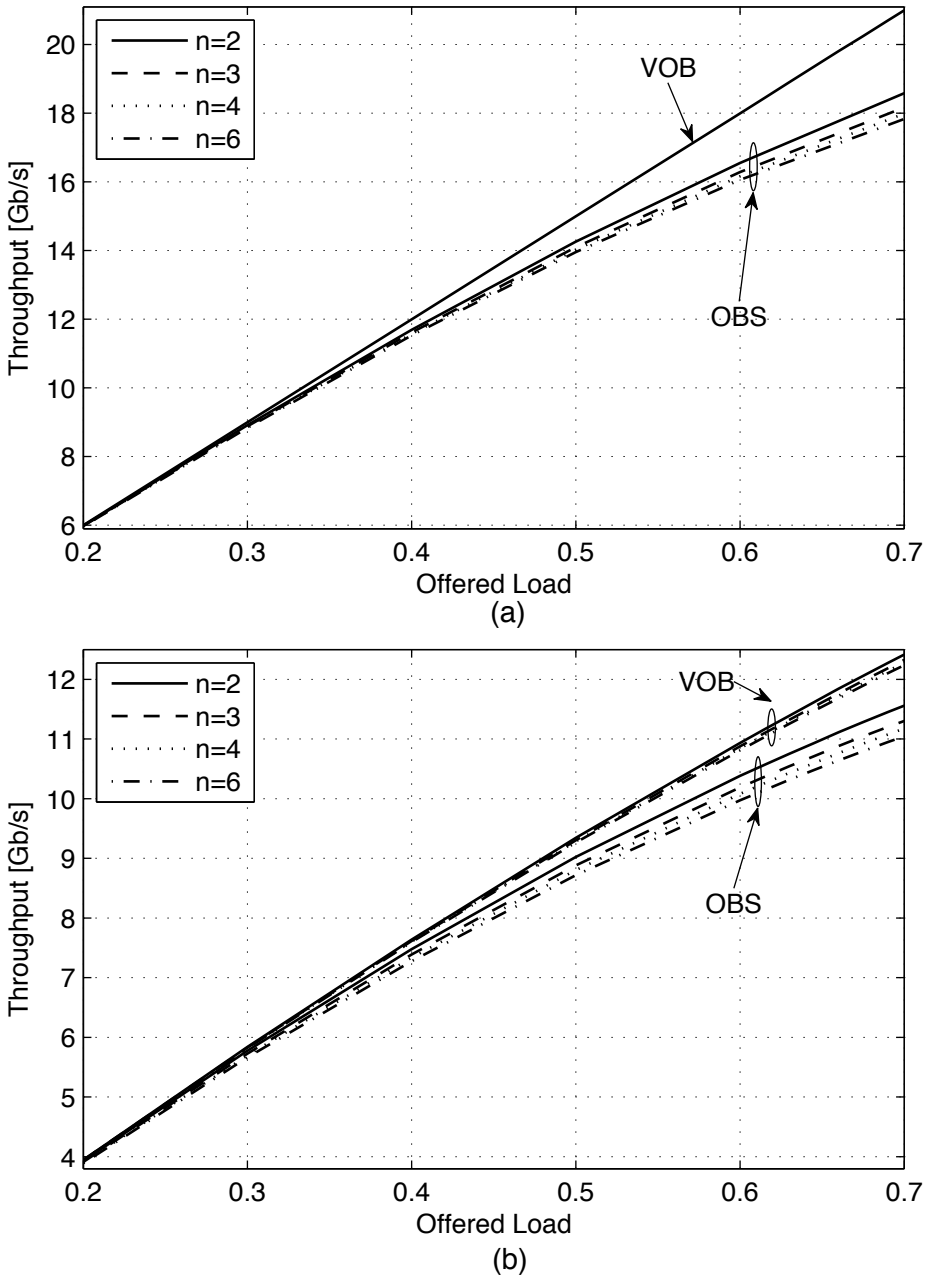

Figure 7.7: Throughput of the network against the offered load to link L for B= 3, (a)W=3, (b) W=2.

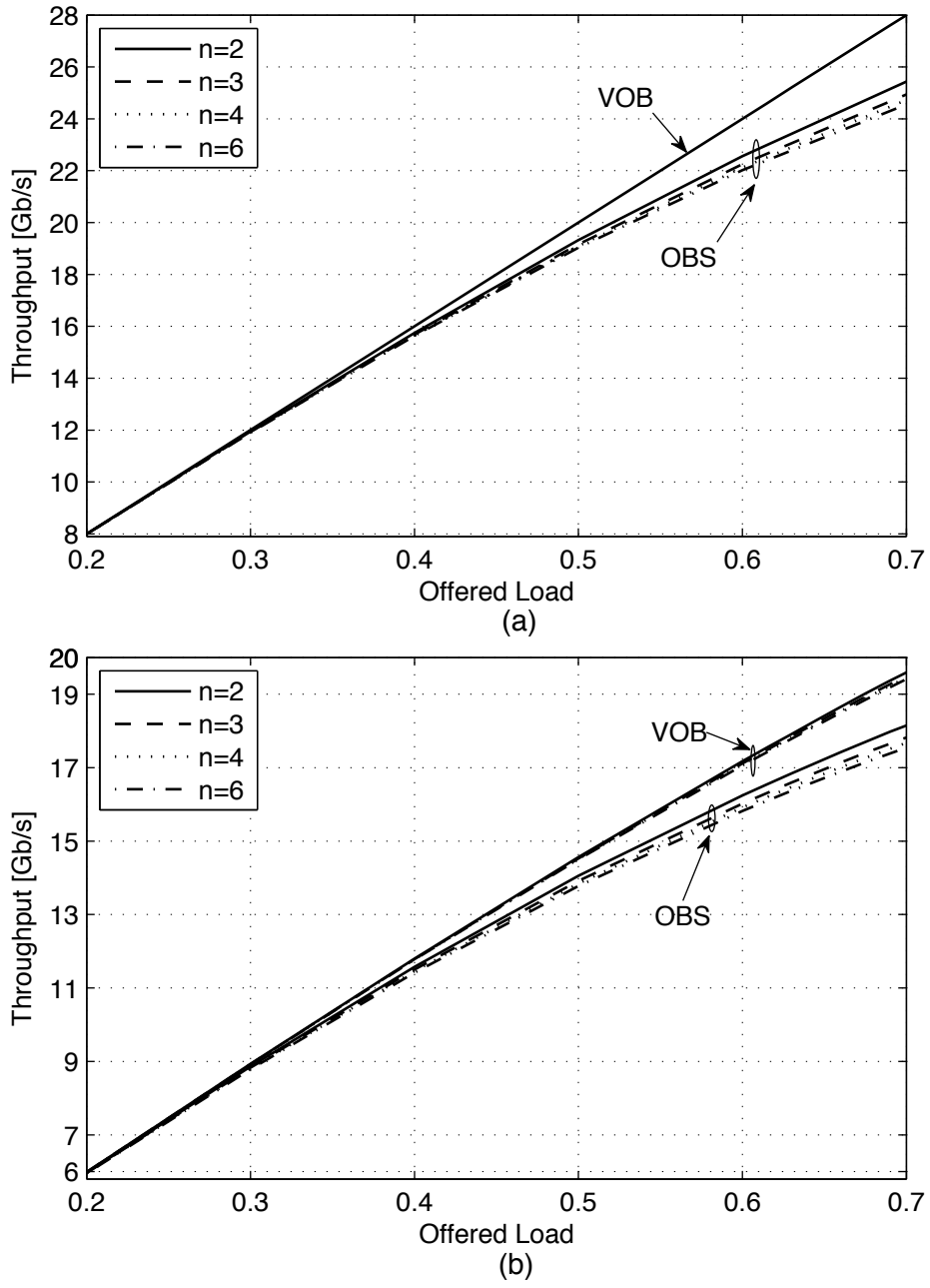

Figure 7.8: Throughput of the network against the offered load to link L for B= 4, (a)W=4, (b) W=3.

other as long as $W_L \geq B$.

In VOB mode II, the access delay is less than 10 μs even at 70-75% VOB load and with 6 upstream nodes. The reason is that for a given number of VOBs to keep the load on the link fixed while decreasing W_L, the load per VOB has to be reduced accordingly. The reduction in the VOB load, in turn, pushes the access delay curve under the knee point, where the access delay is very low.

Overall, it is seen that the VOB shaping delay for all the considered scenarios is acceptable as long as the load offered to the VOB is smaller than a given threshold, which is in the range 70-75% of the single channel capacity. This requirement can be enforced in the design process of the VOB network by properly setting the A_{max} value, which limits the load offered to any VOB in the network.

7.2.3 Impact of Burst Assembly

Up to this point we have investigated the performance of VOB MAC protocol under the assumption that the traffic generated at each source node follows the exponential burst interarrival times and fixed burst sizes. In fact, in Chapters 4 and 5, it was shown that using this assumption in evaluating the classical OBS network results in a lower bound on the performance of the network. In this section we investigate impacts of using assembled burst traffic on the performance of the VOB MAC protocol.

We consider the same network setting as described at the beginning of this chapter, except that here each source node generates traffic according to the hybrid burst assembly algorithm. The length threshold of the assemblers is set to L_{Th}=10 KB and the time threshold is varied in a way that different values of P_{Tout}=0, 0.25, 0.5 and 0.75 are achieved (see Section 4.4). To implement these changes in our model we use the algorithms developed in Section 4.3.

Our simulation analysis have shown that, as expected, using the assembled burst traffic does not have a notable impact on the burst loss rates and the throughput of the considered VOB network. It comes at no surprise since there is actually no buffering for contention resolution in the network and therefore shaping the traffic through the burst assembly does not have a considerable impact on the loss rate. This is, however, not the case for the VOB access delay. Therefore, some sample results of VOB access delays are presented in Figs. 7.13 and 7.14 for the case where B=3.

There are two issues that contribute to the differences between the delay curves in Figs. 7.13 and 7.14 : burst interdeparture time and burst length distributions. In fact, the burst assembly makes the burst arrival times be smoother than Poisson and on the other hand increases the variations of burst sizes as compared to the fixed burst size. Also, we recall from Section 4.4 that the variation of burst size increases and that of burst interdeparture times decreases with P_{Tout}. The combination of these two makes the VOB access delay be smaller under the assembled burst traffic than that under the Poisson traffic before the knee point. On the other hand, when the load increases beyond the knee point for the assembled burst traffic, the increase in the

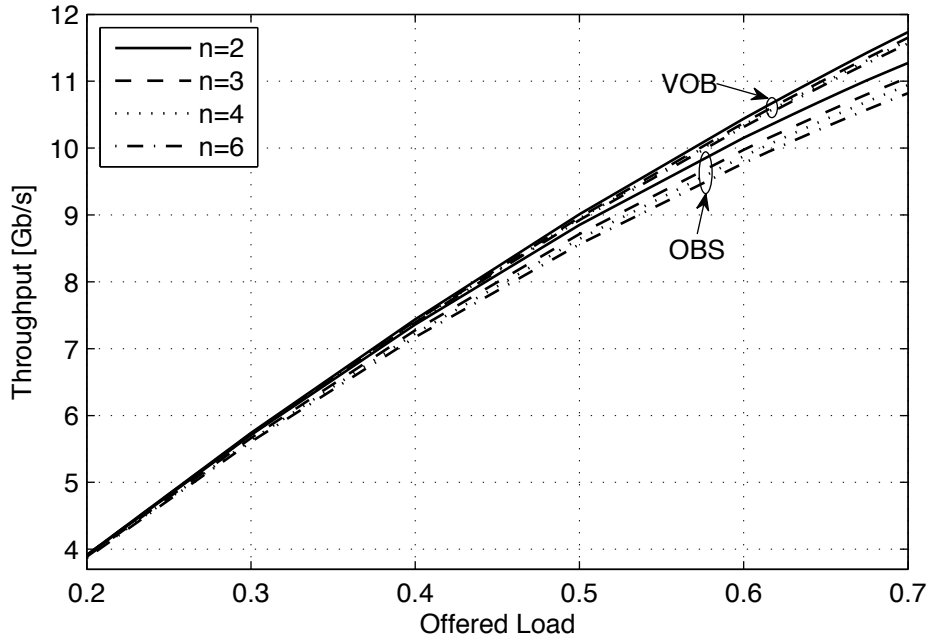

Figure 7.9: Throughput of the network against the offered load to link L for B= 4 and W=2.

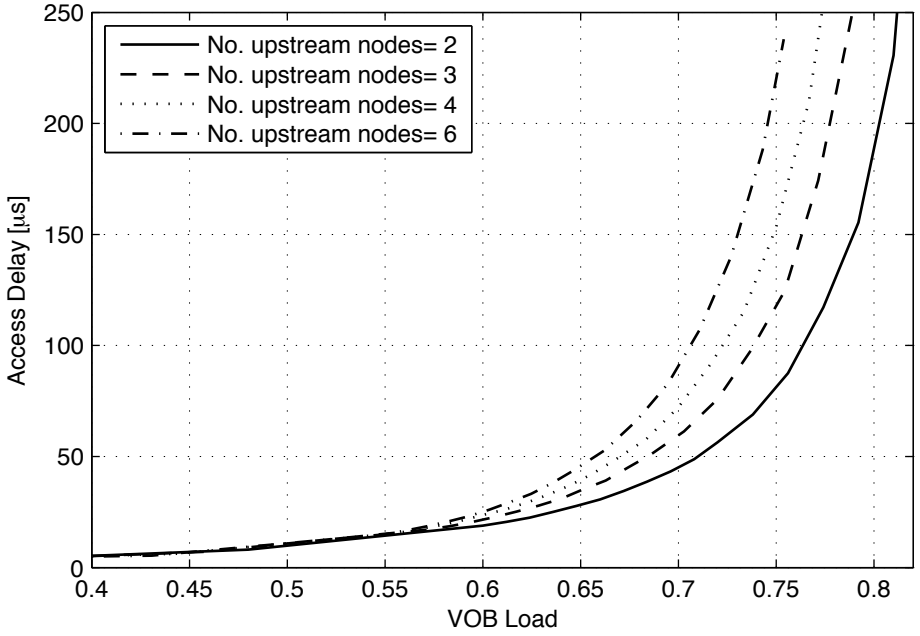

Figure 7.10: Access delay experienced by node $S_{1,1}$ in accessing VOB 1 at B=2 and W=2.

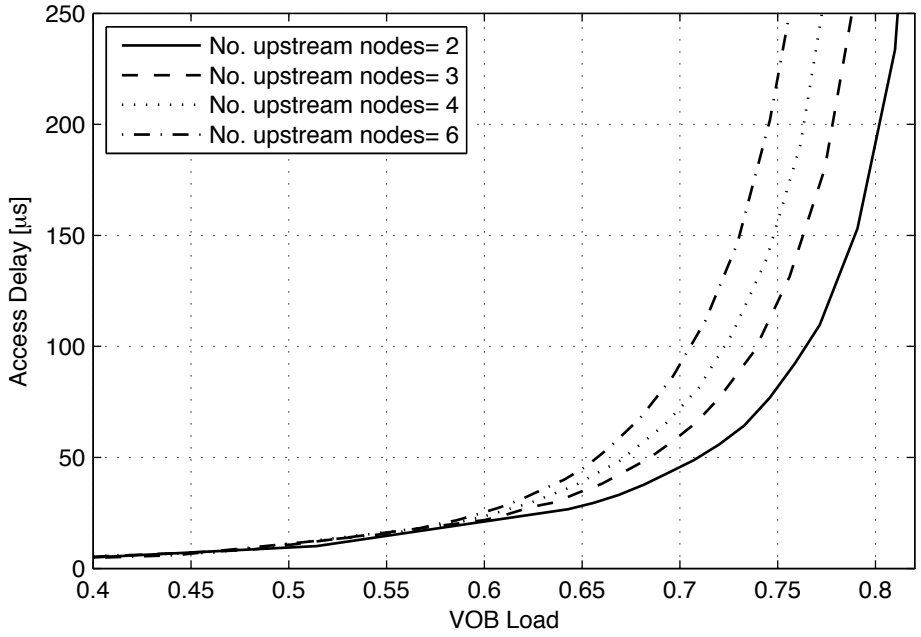

Figure 7.11: Access delay experienced by node $S_{1,1}$ in accessing VOB 1 at B=3 and W=3.

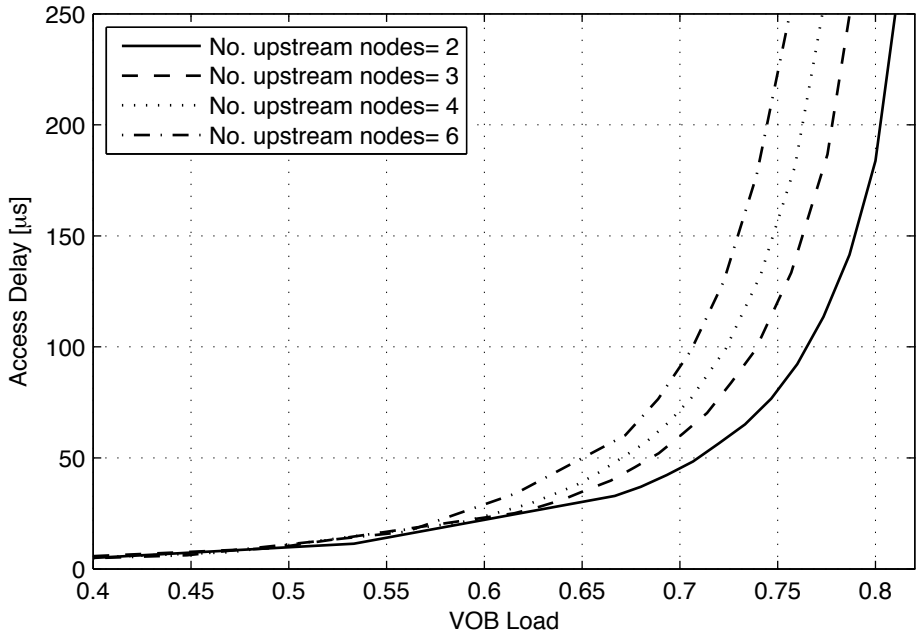

Figure 7.12: Access delay experienced by node $S_{1,1}$ in accessing VOB 1 at B=4 and W=4.

VOB access delay is sharper than that for the Poisson traffic. Overall, we observe that up to a VOB load around 70-75% using the poisson traffic model with fixed burst size results in an upper-bound on the VOB access delay as compared to the assembled burst traffic. Additionally, comparing the access delay curves under assembled burst traffic at different values of P_{Tout}, we can see that the access delay increases and the knee point is shifted to the smaller load values when P_{Tout} increases.

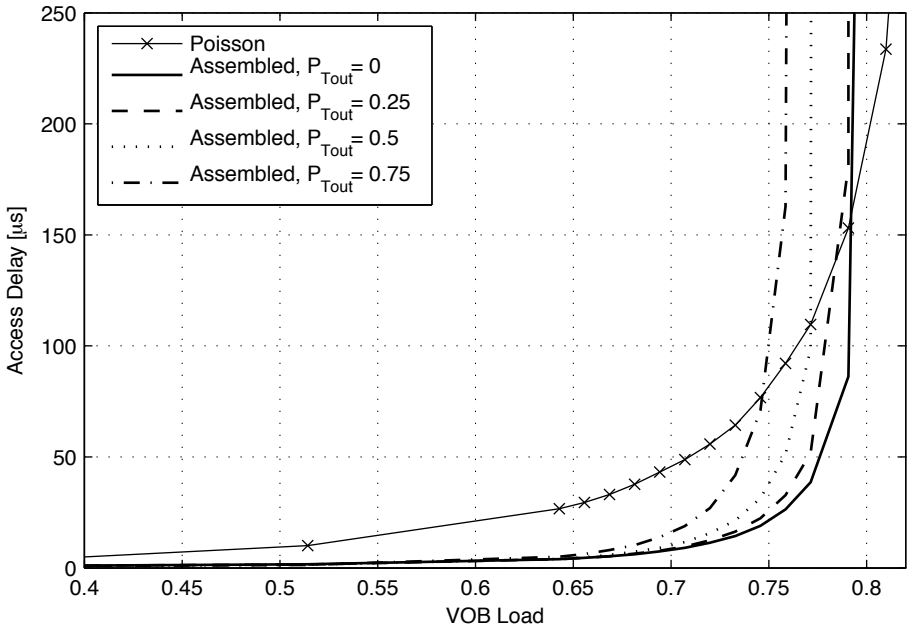

Figure 7.13: Access delay experienced by node $S_{1,1}$ in accessing VOB_1 under different traffic patterns (B=3, n=2 and W=3). Poisson denotes the case where bursts interarrival times are Poisson and burst sizes are fixed at 10 KB. Assembled denotes the case where bursts are generated according to the hybrid burst assembly at L_{Th}=10 KB.

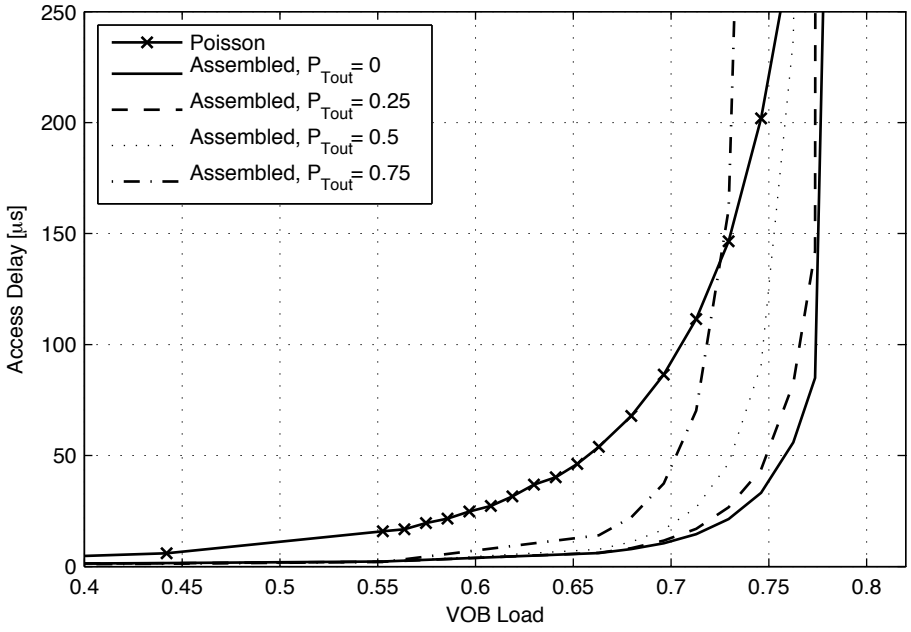

Figure 7.14: Access delay experienced by node $S_{1,1}$ in accessing VOB_1 under different traffic patterns (B=3, n=6 and W=3). Poisson denotes the case where bursts interarrival times are Poisson and burst sizes are fixed at 10 KB. Assembled denotes the case where bursts are generated according to the hybrid burst assembly at L_{Th}=10 KB.

Chapter 8

VOB Network Design Examples

In this chapter we present two design examples of the VOB network and evaluate performance of designed networks. For this purpose, we consider two network topologies as explained below.

In the first design example, we consider a network with 10 nodes, where nodes are connected according to a bidirectional ring topology. Each node is connected to any of the two adjacent nodes via two WDM fibers in different directions, i.e., there are 20 fiber links in the network.

In the second example, the NSFNET backbone network with 14 nodes are considered, as depicted in Fig. 8.1. The network has 42 unidirectional fiber links and the minimum and maximum nodal degree of the nodes are 2 and 4, respectively.

In both networks it is assumed that each WDM link supports one control channel and four data channels, where each data channel operates at rate 10 Gb/s. The architecture of the nodes is the same as the one described Chapter 2. In the ring network, each node is equipped with $T_n = R_n = 8$ tunable lasers and receivers as well as $G_n = 8$ internal wavelength converters. For the NSFNET, the number of available tunable lasers and receivers as well as that of wavelength converters in node i ($0 \leq i \leq 13$) are set to $T_n = R_n = G_n = 4 \times ND_i$, where ND_i is the nodal degree of node i.

8.1 Solving the ILP

8.1.1 Ring Network

For the ring network described above we generate two different traffic matrices. The first one is a random traffic matrix as depicted in Table 8.1. Each element in the matrix of Table 8.1 represents the average traffic demand between two corresponding nodes (i.e. a single O-D flow), as normalized to the capacity of a single wavelength. To generate the traffic matrix, we used a uniform random generator and scaled the generated values such that the maximum O-D demand across the network is limited to 0.4. This selection provides possibilities for grouping O-D flows into VOBs. The

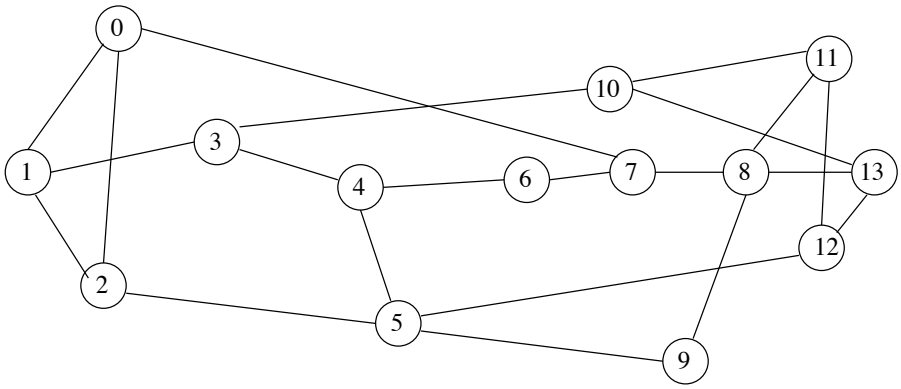

Figure 8.1: NSFNET topology.

average demand per O-D flow for the generated traffic matrix is equal to 0.187. The second traffic matrix is a uniform one, where the demands of all O-D flows are the same and equal to 0.187. That is, the average demand per O-D flow is the same for both the traffic matrices. In the pre-processing phase both possible paths[1] for each source-destination pair are included $\mathbb{P}$. The commercial solver CPLEX 9.0 [172] is used to solve the ILP problem formulated in Chapter 6. Our analysis of the VOB in Chapter 7 suggests that the parameter A_{max} should be set to a value around 0.7 in order that all the traffic sources associated to a VOB experience a reasonable access delay. Accordingly, the ILP problem is solved for each of the two traffic matrices and with three values of the parameter A_{max}; 0.6, 0.7 and 0.75. Summaries of the relevant characteristics of resulting VOB networks and a comparison with the corresponding OBS network are depicted in Figs. 8.2 and 8.3. Details of the solutions are given in Table A.1-A.5 in Appendix A. Further processing of the detailed solutions shows that for the considered scenarios the required number of insertion buffers per node in the network, on average, varies between 4.1 and 4.8, which is much less than the theoretical upper bound given in Chapter 6.

We observe that the number of VOBs associated to the links in the VOB network for different settings is much smaller than the number of flows per link in the OBS network. More specifically, the average number of VOBs associated to a link in the network varies between 3.95 and 4.95, whereas in the OBS architecture 12.5 independent flows are associated to each link, on average. Recall that our goal is to minimize the number of VOBs (flows) per link in the network in order to reduce burst contentions. Also, comparing the loads per link for different settings in Figs. 8.3, we see that the VOB does not affect the average traffic load assigned to each link in the network noticeably, which implies that the routing of the traffic in the considered VOB network is not effectively different from the shortest path routing applied in OBS.

[1]There are two candidate paths for each source-destination pair on a bidirectional ring topology.

Table 8.1: Random traffic matrix for the ring network used in the design example. The values are normalized to 10 Gb/s.

	0	1	2	3	4	5	6	7	8	9
0	0.000	0.180	0.043	0.173	0.341	0.167	0.312	0.094	0.219	0.372
1	0.318	0.000	0.385	0.364	0.249	0.020	0.156	0.141	0.119	0.310
2	0.124	0.092	0.000	0.073	0.140	0.361	0.097	0.328	0.298	0.195
3	0.211	0.365	0.310	0.000	0.205	0.378	0.162	0.006	0.076	0.174
4	0.066	0.061	0.327	0.058	0.000	0.196	0.039	0.017	0.275	0.179
5	0.241	0.330	0.347	0.054	0.030	0.000	0.053	0.068	0.073	0.123
6	0.105	0.215	0.034	0.348	0.096	0.135	0.000	0.260	0.147	0.203
7	0.262	0.398	0.160	0.232	0.049	0.360	0.382	0.000	0.250	0.204
8	0.276	0.031	0.104	0.220	0.074	0.148	0.230	0.259	0.000	0.327
9	0.299	0.177	0.320	0.058	0.096	0.044	0.024	0.180	0.032	0.000

Now let us consider the difference between the solutions at different values of A_{max}. First, we observe that under the random traffic matrix, increasing A_{max} from 0.6 to 0.7 results in a reduction of the number of VOBs per link, however, increasing A_{max} beyond 0.7 does not affect the number of VOBs per link. This indicates that the impact of A_{max} on the average number of VOBs per link is a step-wise relation, that is, there is a range of values of A_{max} over which the average number of VOBs per link remains unchanged. On the other hand, although setting the A_{max} to any value within this range does not affect the number of VOBs per link, we expect that it impacts the performance of the network. For example, in the considered scenarios the maximum number of VOBs per link in the network is 4 at both values of A_{max}=0.7 and 0.75 for the random traffic matrix, and as each link in the network supports 4 wavelength channels, the loss rate would be zero in the network at both values of A_{max}. However, we expect the access delay to be higher for $A_{max} = 0.75$ as the VOBs are more packed in this case compared with the case of $A_{max} = 0.7$. As a conclusion, it would be better – in terms of the access delay – to design the VOB layout of a network at the minimum value of the A_{max} in the interval where the average number of VOBs per link remains fixed.

In contrast to the case with random traffic, under the uniform traffic matrix varying the A_{max} around 0.7 causes larger variation in the number of VOBs per link. In fact, under the uniform traffic matrix all the demands are the same and therefore the play ground for the optimization and for clustering is not as large as that under the random traffic. For example, for the uniform traffic considered in this analysis the maximum number of O-D flows that can be supported on a single VOB on a link is limited to 4 and 3 for A_{max}=0.7 and 0.75, respectively, and if the number of flows is increased beyond these values the corresponding A_{max} will be violated. That is, at A_{max}=0.7

Table 8.2: Traffic matrix of NSFNET. The values are normalized to 10 Gb/s.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	0.00	0.08	0.03	0.01	0.06	0.00	0.02	0.07	0.01	0.01	0.09	0.03	0.06	0.01
1	0.21	0.00	0.18	0.09	0.17	0.08	0.12	0.46	0.03	0.06	0.24	0.31	0.23	0.16
2	0.03	0.14	0.00	0.14	0.03	0.11	0.03	0.26	0.03	0.01	0.15	0.02	0.06	0.04
3	0.02	0.02	0.04	0.00	0.01	0.00	0.00	0.01	0.01	0.01	0.04	0.04	0.02	0.01
4	0.37	0.48	0.06	0.01	0.00	0.01	0.03	0.19	0.07	0.05	0.22	0.35	0.06	0.04
5	0.01	0.05	0.01	0.02	0.01	0.00	0.01	0.01	0.00	0.01	0.02	0.01	0.02	0.00
6	0.11	0.19	0.31	0.01	0.07	0.02	0.00	0.34	0.06	0.07	0.46	0.28	0.49	0.07
7	0.04	0.70	0.63	0.03	0.08	0.01	0.29	0.00	0.13	0.10	0.27	0.21	0.27	0.06
8	0.25	0.06	0.11	0.02	0.07	0.02	0.07	0.18	0.00	0.12	0.33	0.44	0.19	0.14
9	0.01	0.13	0.03	0.01	0.07	0.03	0.01	0.17	0.02	0.00	0.11	0.08	0.04	0.04
10	0.03	0.11	0.17	0.02	0.03	0.04	0.06	0.11	0.06	0.07	0.00	0.18	0.11	0.10
11	0.09	0.39	0.06	0.04	0.13	0.02	0.05	0.17	0.12	0.09	0.63	0.00	0.20	0.08
12	0.24	0.68	0.16	0.07	0.27	0.09	0.10	0.27	0.09	0.00	0.39	0.41	0.00	0.19
13	0.12	0.17	0.06	0.02	0.03	0.00	0.01	0.07	0.35	0.11	0.21	0.24	0.16	0.00

a VOB could be loaded at most to $3 \times 0.187 = 0.561$ of the channel capacity; thus number of required VOBs to support the traffic demand at A_{max}=0.7 would be much larger than that at A_{max}=0.75.

8.1.2 NSFNET

The traffic matrix used for NSFNET topology is a measurement-based estimation of traffic in the NSFNET backbone as given in [6]. The estimation presented in [6] shows the aggregate number of bytes between each pair of nodes in a 15-minute interval. We use these numbers as an approximate indication of relative intensities of demands between pairs of nodes in the network and accordingly scale the numbers in a way that the maximum element of the demand matrix is equal to 7 Gb/s. Note that if the demand between any two nodes in the network is larger than around 70% of the capacity of a single wavelength, it might make more sense to establish a direct point-to-point light-path between the two nodes. The resulting demand matrix is depicted in Table 8.2, where each value is normalized to 10 Gb/s.

In contrast to the ring topology, here the total number of possible paths that can be generated by the k-shortest path routing algorithm in the preprocessing phase is not a small number. Therefore, in order to evaluate possible impacts of total number of paths on results of optimization, we solve the ILP with four sets of VOB candidates $\mathbb{P}$, where each has a different size P. Taking into account that in the NSFNET with 14 nodes the set $\mathbb{P}$ should at least contain 182 paths, i.e. one path for each source-destination pair, we solve the ILP formulation for $P = 182, 364, 546, 728$, which correspond to 1, 2, 3 and 4 paths per source-destination pair, respectively. For solving the ILP in all

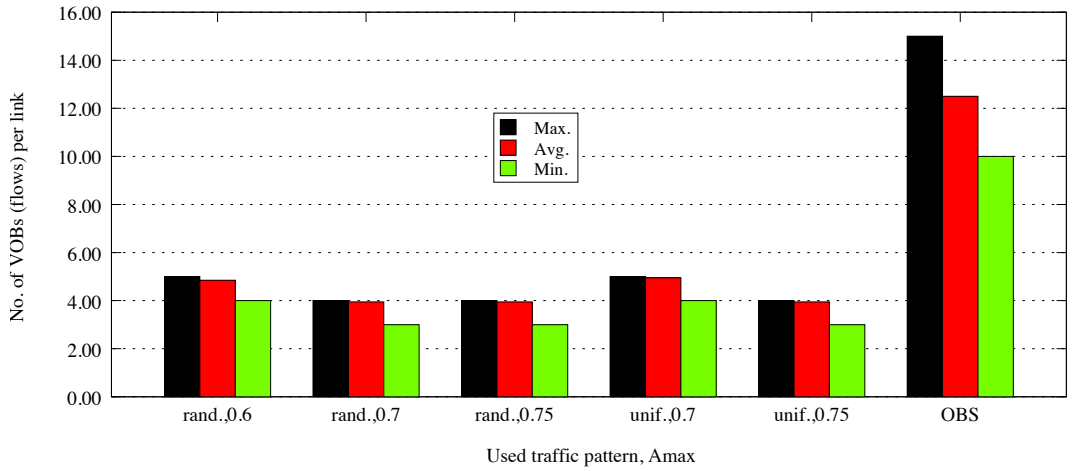

Figure 8.2: Number of VOBs per link resulted from solving the ILP for the ring network at different traffic matrices (random and uniform) and at different values of A_{max}. The results are compared with the classical OBS network for which the number of independent flows per link are shown.

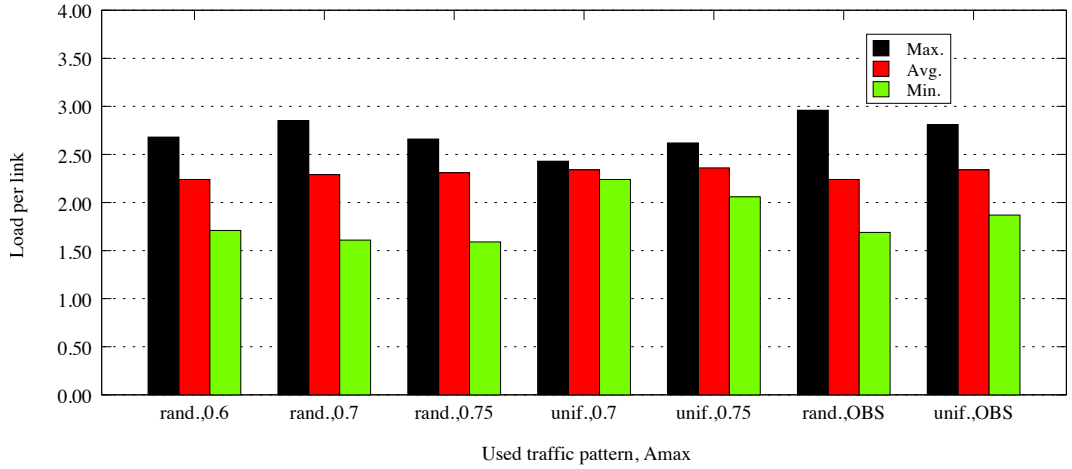

Figure 8.3: Load per link resulted from solving the ILP for the ring network at different traffic matrices (random and uniform) and at different values of A_{max}. The results are compared with the classical OBS network. Load values are normalized to the capacity of a single wavelength channel.

the considered cases, we set $A_{max} = 0.7$. Summaries of the results obtained through solving the corresponding ILPs are depicted in Figs. 8.4 and 8.5. Also, details of the solutions are given in Table A.6-A.9 in Appendix A. As for the routing of the flows, the detailed solutions show that, as expected, the asymmetry between the routes for flows with the same end points increases with k, such that at k=4 around 56% of bidirectional flows take different paths on opposite directions. Furthermore, processing of the solutions demonstrates that, on average, only 3.93 to 5.43 insertion buffers are required per node for different values of k.

First observe that, as expected, applying the clustering significantly reduces number of independent flows per link in the VOB network as compared to the classical OBS network. For instance, at k=4 the maximum number of independent flows per any link in the VOB network is equal to 4, whereas it is equal to 18 in the OBS network, i.e. around 78% reduction is observed. In addition, it is seen that even establishment of VOBs based on the set of paths obtained from the first shortest path routing, the case with k=1, considerably reduces both total number of independent flows and number of flows per link.

As for the impact of k on the optimization results, it is seen that at the beginning, increasing k from 1 to 2 leads to a sharp reduction in the number of VOBs per link. Nevertheless, beyond k=2 further improvements of the results through increasing k slows down. This is in fact because the optimum value of the objective function is approached. More specifically, the optimum value of the objective function is achieved at k=3 and increasing of k will not improve the solution any further. Nevertheless, increasing k from 3 to 4 leads to the slightly smaller number of total VOBs in the network since it results in the selection of longer of VOBs by the solver. This is also reflected in the increased value of the average number of VOBs per link at k=4 as compared to that at k=3.

8.2 Performance Results

To quantify the performance of the VOB networks explained in the last section we use a simulation model developed in OMNeT++. The simulation model can be set to work under VOB or OBS. Packets for each O-D flow are generated according to the Poisson process with deterministic packet size of 10 KB. The assumptions on the traffic are in line with the traffic analysis presented in Chapters 4 and 5 and also with the analysis of Chapter 7. For the ring network the arrival rates of the packets are set in a way that the average load generated for each flow is in agreement with the corresponding value in Table 8.1 for the random traffic scenario and is equal to 0.187 for the uniform traffic matrix. Similarly for the NSFNET scenario, the arrival rates are set according to load values given in Table 8.2. For the ring network under uniform traffic we consider two scenarios: without FDL buffers (F = 0) and with a single FDL buffer per output port (F = 1). In case F=1, the length of the delay line is set in a way that it provides

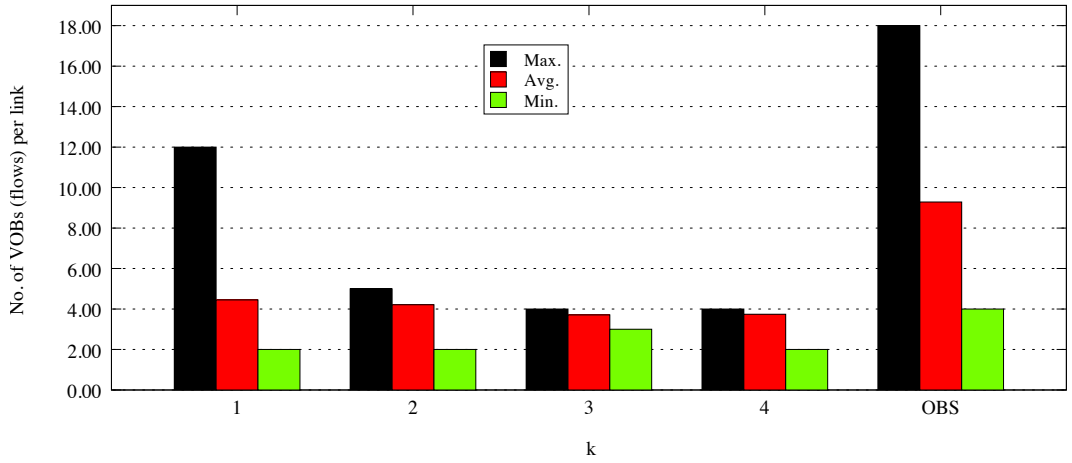

Figure 8.4: Number of VOBs per link resulted from solving the ILP for the NSFNET topology at A_{max}=0.7 and at different values of k. The results are compared with the classical OBS network for which the number of independent flows per link are shown.

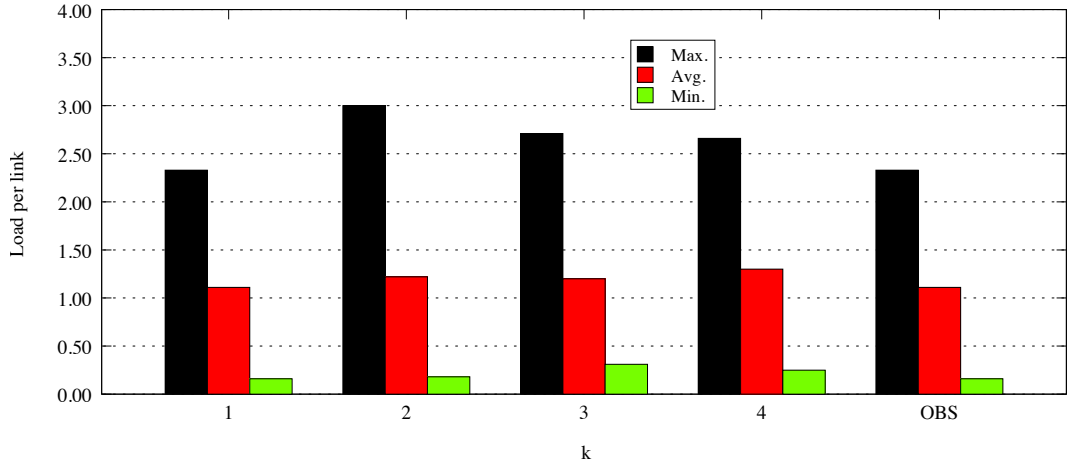

Figure 8.5: Load per link resulted from solving the ILP for the NSFNET topology at A_{max}=0.7 and at different values of k. The results are compared with the classical OBS network. Load values are normalized to the capacity of a single wavelength channel.

delay equal to the transmission time of a single packet over the channel. For the ring network under random traffic as well as the NSFNET scenario, we only consider the case without FDL buffers. Other settings at the nodes are the same as those explained in Chapter 7. For the sake of simplicity, in our simulations we assume the propagation delays in the network to be negligible. Additionally, in case of OBS, the shortest path routing is used for routing packets through the network.

Tables 8.3 and 8.4 present the results of simulating the ring network and the NSFNET, respectively. Shown in the tables are the average values of packet drop rate, throughput and access delay as well as the maximum value in the set of the average access delays of all the O-D flows for each scenario. The results are compared with those obtained from the simulation of OBS.

We observe that the packet drop rate improves greatly when the VOB architecture is employed in comparison to OBS. Specifically, for the case that the maximum number of VOBs per link is equal to the available wavelength channels on the link, the loss rate is completely suppressed and no instance of packet loss is observed during the long simulation periods. Also, for the case that number of wavelength channels are smaller than that of VOBs, a large reduction – up to two orders of magnitude – in the loss rate is observed.

The elimination/reduction of the loss rate, which has been the first objective of designing the VOB architecture, gives rise to the improvement of the overall throughput of the network. For the considered ring network the throughput improvement is up to 13.57%, which is associated to the uniform traffic with A_{max}=0.75 and without FDL buffer. Also for the case of NSFNET, around 1.7% increase in the overall network throughput is achieved. As observed, the improvements in the throughput for the NSFNET scenario as well as the ring network at F=1 is marginal. The reason is that for these cases the fraction of packets that are dropped in the network under OBS is of the order of 10^{-2}–10^{-3}, i.e., there is marginal room for improvement in the throughput. It should be, however, noted that although a good throughput is achieved for these cases under classical OBS, the drop rate is not acceptable taking into account that each – dropped – burst is the result of aggregating several packets, which can be consecutive packets of a single TCP flow.

Table 8.4 also includes the results achieved at different values of k, which in turn reflect impacts of routing of O-D flows on the performance of VOB architecture. For instance, at k=1 the same routing is applied in VOB network as in OBS. From the table it is clear that routing of flows has an important role in the VOB architecture.

The cost of improvement in the packet loss rate and the network throughput is the access delay penalty that is introduced by the flow coordination in the VOB approach. Nevertheless, it is seen in Tables 8.3 and 8.4 that the delay penalty is negligible for all the investigated scenarios–taking into account that propagation delays in a typical metro/core network are in the range of tens of ms. For instance the average length of the links in NSFNET is 1081 km, which leads to a 5.4 ms propagation delay per hop, on average. The acceptable access delays indicate that the values of A_{max} are set

Table 8.3: Performance evaluation results for the ring network. The average values are shown with 90% confidence level.

			Avg. Packet Drop Rate	Avg. Network Throughput (Mb/s)	Avg. Access Delay over all flows (μs)	Max Access Delay over all flows (μs)
Random Traffic	F=0	VOB A_{max}=0.6	3×10^{-2} $\pm 1.5 \times 10^{-5}$	163,225 ± 18	5.91 ± 0.1	16.83 ± 0.13
		VOB A_{max}=0.7	0 -	168,645 ± 5.02	12.3 $\pm 6.5 \times 10^{-3}$	52.7 ± 0.77
		VOB A_{max}=0.75	0 -	168,645 ± 4.7	17.01 $\pm 5 \times 10^{-2}$	111 ± 5.1
		OBS	9.4×10^{-2} $\pm 2.4 \times 10^{-5}$	152,763 ± 8.9	1.7 $\pm 3.7 \times 10^{-4}$	2.98 ± 0.1
Uniform Traffic	F=0	VOB A_{max}=0.7	3.6×10^{-2} $\pm 2.4 \times 10^{-5}$	162,287 ± 26.2	6.2 $\pm 7.5 \times 10^{-3}$	12.2 ± 2.2
		VOB A_{max}=0.75	0 –	168,657 ± 32	21.1 $\pm 4.9 \times 10^{-2}$	62.7 ± 11.5
		OBS	1.2×10^{-1} $\pm 6.6 \times 10^{-5}$	148,250 ± 25.2	1.2 $\pm 7.5 \times 10^{-3}$	1.8 ± 1.3
	F=1	VOB A_{max}=0.7	5.9×10^{-5} $\pm 6.4 \times 10^{-7}$	168,561 ± 7.1	6.6 $\pm 1.6 \times 10^{-3}$	12.6 ± 2.3
		OBS	3.8×10^{-3} $\pm 1.7 \times 10^{-5}$	167,732 ± 30.3	2.1 $\pm 3.5 \times 10^{-3}$	4.2 $\pm 6.5 \times 10^{-1}$

properly in the VOB layout design phase.

To sum up, it is observed that in the VOB network design there is a trade-off between the edge shaping delay, i.e. the VOB access delay, and the loss rate inside the network. In the extreme case where there is no clustering in the network, we expect that the performance of the network approaches that of the classical OBS. In this case, the VOB MAC protocol will perform very similar to OBS since there will be no need for checking the insertion buffer before injecting the packets into the network. When clustering is introduced in the network, on one hand burst losses reduce, while on the other hand, access delays increase, where the amount of these changes in the loss rate and access delay depends, among others, on how dense the clusters are. In fact, allowing larger shaping delays at the edge of the network, realized by increasing the A_{max}, results in the formation of denser clusters and thereby in further reduction of burst losses. Moving in this direction – depending on the load distribution and the available network resources – a network operation point is reached where the maximum number of VOBs per link is equal to the number of channels par link and therefore the loss rate inside the network is fully suppressed. Beyond this point, making the VOBs denser results merely on the increased shaping delay.

Table 8.4: Performance evaluation results for the NSFNET. The average values are shown with 90% confidence level.

	Avg. Packet Drop Rate	Avg. Network Throughput (Mb/s)	Avg. Access Delay over all flows (μs)	Max Access Delay over all flows (μs)
VOB (k=1)	1.08×10^{-2} $\pm 2.42 \times 10^{-5}$	217,022 ± 27.19	3.74 $\pm 4.9 \times 10^{-3}$	20.11 ± 0.16
VOB (k=2)	6.6×10^{-3} $\pm 7.47 \times 10^{-6}$	217,947 ± 27.52	5.86 $\pm 4.5 \times 10^{-3}$	43.69 ± 0.73
VOB (k=3)	0 -	219,400 ± 28.18	6.75 $\pm 8.8 \times 10^{-3}$	49.65 ± 0.92
VOB (k=4)	0 -	219,400 ± 28.19	6.93 $\pm 1 \times 10^{-2}$	42.07 ± 0.43
OBS	1.75×10^{-2} $\pm 2.01 \times 10^{-5}$	215,567 ± 17.4	2.18 $\pm 1.7 \times 10^{-3}$	9.41 $\pm 2.7 \times 10^{-2}$

8.3 Impact of Load Variations

In the design and performance evaluation of the VOB networks in the last sections we assumed that the traffic matrix is fixed. However, in reality the traffic matrix changes over time, which might necessitate redesign of the VOB network layout in order to achieve desired performance. Therefore, in this section we investigate, by means of some examples, how changes in the traffic matrix can affect the performance of the VOB network.

For this purpose, we consider the ring network that was introduced earlier in this chapter. The demand matrix given in Table 8.1 is taken as the initial traffic matrix. Based on this, new traffic matrices are produced, which differ from the initial one – in terms of the load intensity – between -20% to +20%. More specifically, the initial traffic matrix is multiplied by 0.8, 0.9, 1.1 and 1.2 to generate traffic matrices with -20%, -10%, +10% and +20% difference, respectively. For each new traffic matrix, the performance of the ring network is evaluated under three settings: i) the VOB network is designed for the new traffic matrix, ii) the VOB network is designed for the initial traffic matrix and iii) the network operates under classical OBS.

First let us consider the design of the VOB network for the new traffic matrices. A summary of characteristics of the resulting VOB networks are presented in Figs. 8.6 and 8.7. Details of the solutions are given in Table A.10-A.15 in Appendix A. As expected, for a given A_{max} value, both the number of VOBs per link and the load per link increase or decrease in the same direction that the traffic matrix changes.

The results of simulating the ring network with the new traffic matrices under different settings are presented in Table 8.5. First we consider the case that the load increases. In this case, the burst loss rate is smaller in the network designed for the

initial traffic matrix than in the network designed for the new traffic matrix. For instance at A_{max}=0.7 with +20% load change, the loss rate is zero in setting II, while it increases to 7.1×10^{-2} in setting I. On the other hand, for the same scenario the maximum VOB access delay in setting II is around one order of magnitude larger than that in setting I. To explain this behavior we recall that in the network designed for the initial traffic matrix, the maximum number of VOBs per link is equal to the number of wavelength channels per link (see Fig. 8.2), which means no loss rate in the network. Accordingly, when the load increases in this network we still have the zero loss rate, since the VOB layout does not change in the network. Instead, the effect of the load increase is manifested in the delay, because load per VOB violates the A_{max} value that the network has been designed for. Consequently, when the desired A_{max} value is enforced by redesigning the network based on the new traffic matrix, the maximum number of VOBs per link increases to 5 (Fig. 8.6), which results in a reduction of the VOB access delay at the cost of introducing some burst loss in the network. That is, as discussed in the last section, there is a trade-off between the VOB access delay and the burst loss rate in the network.

Following the same line of reasoning, three different situations can be identified when the load in the network scales down. The first situation occurs when the burst loss rate is zero under the initial traffic matrix, i.e. the maximum number of VOBs per link is equal to or smaller than the number of available wavelength channels on the links. An example of this case is VOB network at A_{max}=0.7 in Table 8.5. In this situation the reduction of the load in the network designed for the initial load has no impact on the loss rate, but reduces the VOB access delay. Therefore, redesigning the VOB layout is not necessary in this case. The second situation occurs when there are already burst losses in the network under the initial traffic matrix and the reduction in the load is so large that redesigning the VOB layout of the network can result in a smaller number of VOBs per link and thereby can fully suppress the burst loss in the network. In this case, further operation of the network under the layout designed for the initial traffic results in burst losses, which would be avoided if the VOB network is redesigned based on the new traffic matrix. Therefore in this case, redesigning the network is recommended. An example of this case is VOB at A_{max}=0.6 in Table 8.5. The third case is the same as the second one, except that if the network is redesigned the change in load reduction is not large enough to reduce the maximum number of VOBs per link. Again, it is not necessary in this case to redesign the network.

Finally, as depicted in Table 8.5 we observe that for the new traffic matrices, the operation of the VOB network, which has been designed for the initial traffic matrix, is always better than classical OBS. Though as pointed out before, this comes at a cost of increased access delay. This, together with the above discussion, suggests that the VOB network can safely handle small variations – in the range ± 10-20% – in load without requiring to change the designed layout.

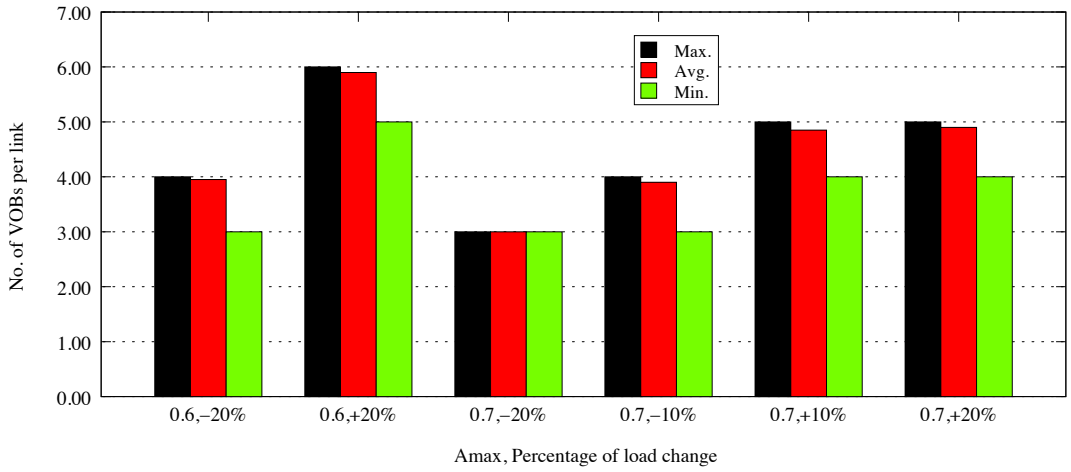

Figure 8.6: Number of VOBs per link resulted from solving the ILP for the ring network at different values of A_{max} and at different load changes with respect to the initial traffic matrix.

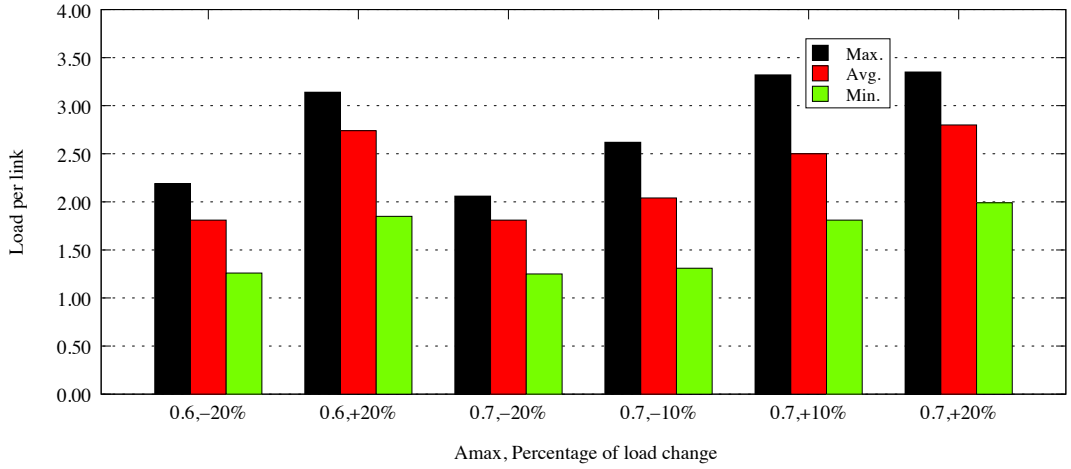

Figure 8.7: Load per link resulted from solving the ILP for the ring network at different values of A_{max} and at different load changes with respect to the initial traffic matrix.

Table 8.5: Performance Evaluation Results for the ring network at different values of A_{max} and at different load changes with respect to the initial traffic matrix. Settings I and II denote the cases where the network is designed for the new traffic matrix and initial traffic matrix, respectively. The average values are shown with 90% confidence level.

	Load change	Setting	Avg. Packet Drop Rate	Avg. Network Throughput (Mb/s)	Avg. Access Delay over all flows (μs)	Max Access Delay over all flows (μs)
VOB $A_{max} = 0.7$	-20%	I	0 -	134,693 ± 16	15.31 ± 0.01	40.9 ± 0.15
		II	0 -	134,693 ± 16	9.64 ± 0.15	33.5 ± 0.22
	-10%	I	0 -	151,530 ± 18	11.3 $\pm 7 \times 10^{-3}$	27.4 ± 0.15
		II	0 -	151,530 ± 18	11.45 ± 0.12	45.2 ± 0.83
	+10%	I	4.4×10^{-2} $\pm 3.3 \times 10^{-5}$	176,899 ± 20	10.14 $\pm 7 \times 10^{-3}$	33 ± 0.13
		II	0 -	185,203 ± 22	19.45 $\pm 1.7 \times 10^{-2}$	90.6 ± 1.2
	+20%	I	7.1×10^{-2} $\pm 4.7 \times 10^{-5}$	187,731 ± 16	10.6 $\pm 8 \times 10^{-3}$	32.46 ± 0.17
		II	0 -	202,040 ± 24	38.27 $\pm 7.6 \times 10^{-2}$	384 ± 8.2
VOB $A_{max} = 0.6$	-20%	I	0 -	134,693 ± 16	6.7 $\pm 4 \times 10^{-3}$	18.2 ± 0.14
		II	1.4×10^{-2} $\pm 1.4 \times 10^{-5}$	132,750 ± 15	3.43 $\pm 3 \times 10^{-3}$	7.6 $\pm 5 \times 10^{-2}$
	+20%	I	9×10^{-2} $\pm 4 \times 10^{-5}$	182,649 ± 18	5.78 $\pm 4 \times 10^{-3}$	13.2 $\pm 7 \times 10^{-2}$
		II	5.4×10^{-2} $\pm 2.9 \times 10^{-5}$	191,071 ± 21	10.85 $\pm 9 \times 10^{-3}$	45.9 ± 0.45
OBS	-20%	III	5.6×10^{-2} $\pm 3.6 \times 10^{-5}$	127,189 ± 13	1.2 $\pm 7 \times 10^{-4}$	2 $\pm 2 \times 10^{-3}$
	-10%	III	7.4×10^{-2} $\pm 3.4 \times 10^{-5}$	140,277 ± 16	1.44 $\pm 7 \times 10^{-4}$	2.4 $\pm 5 \times 10^{-3}$
	+10%	III	0.12 $\pm 4.7 \times 10^{-5}$	163,796 ± 15	2.03 $\pm 1 \times 10^{-3}$	3.44 $\pm 3 \times 10^{-3}$
	+20%	III	0.14 $\pm 4.7 \times 10^{-5}$	174,217 ± 18	2.4 $\pm 1 \times 10^{-3}$	4.12 $\pm 1.1 \times 10^{-2}$

Chapter 9

Conclusion

Transport networks based on WDM technology are among the main building blocks of the Internet, and therefore they have been at the center of many research activities over recent years. The rapid developments in this area have been pushed primarily by two driving forces: the exponential growth of traffic in the Internet and advances in optical technology. The volume of traffic in the Internet has been constantly increasing as a result of the fact that the Internet has turned into an indispensable part of our daily lives, and as an infrastructure it plays a major role in the global developments. This in turn has pushed the need for ultra-high-capacity transport networks that are cost-efficient and enjoy a high scalability. Above all these requirements, the highly varying nature of Internet traffic necessitates that transport networks be highly dynamic and flexible. On the other hand, optical technology in general and WDM in particular have achieved a maturity that can be employed to build transport networks meeting all of these requirements. Consequently, we have been witnessing a variety of proposals and techniques for transport networks aiming at integration of IP and WDM.

One of the promising approaches for integrating IP and WDM is optical burst switching networks. OBS is an optical switching paradigm that facilitates dynamic allocation of networking resources in the optical domain at subwavelength granularity. These features have turned OBS into an attractive choice for realizing next generation WDM transport networks. Nonetheless, OBS networks suffer from burst collisions that can deteriorate operation and the overall performance of networks. The collisions problem arises due to the lack of randomly accessible optical memories. Therefore, effective avoidance and resolution of burst collisions are crucial to the success of OBS networks.

In this dissertation, we aim at improving the performance of OBS networks through devising new contention resolution mechanisms. Specifically, we adopt a novel approach based on traffic shaping at the edge of the network, combined with network-level traffic engineering. For this purpose, we first carry out an in-depth analysis of characteristics of traffic at the ingress edge of OBS networks. In this study, we further investigate the traffic shaping features of burst assembly algorithms at the edge of the network and

their impacts on burst contention inside the network. Following the insights achieved in this study, we put forward a novel set of architecture and protocol for packet-based WDM transport network called Virtual Optical Bus (VOB). We present the design of VOB architecture and demonstrate its high potential in suppressing burst collisions inside a network.

9.1 Summary of Contributions

In this section, we summarize the major contributions made in this work in order of their appearance in this dissertation.

- Exact analytical models are developed for the distributions of size and interdeparture time of bursts that leave the assembly buffer in an ingress OBS node in Chapter 4.
- As an important application of the analytical traffic models developed in Chapter 4, we present synthetic traffic generators to be used in discrete event simulation models. The main benefit of the proposed generators is the large speedup gain that is achieved in simulating OBS networks. In the specific example considered in this dissertation, a speedup gain as high as 40 is observed.
- New insights into the influence of traffic aggregation at the edge of optical networks on the performance of core switching nodes are presented in Chapter 4. Specifically, it is demonstrated that the burst collision rate on a WDM link in the core of OBS network at a given utilization is significantly decreased by reducing the number of flows being multiplexed onto that link, on the condition that traffic of the individual flows are smoothed at the edge. This emphasizes on the potential of employing traffic shaping at the edge of the network to smooth the traffic as much as possible as well as on the applying traffic engineering approaches at the network level to reduce the total number of flows inside the network.
- A comprehensive case study on the burst assembly process in OBS networks using real packet traces collected from Internet is presented in Chapter 5. The study contributes to the understanding of the statistical features of burst traffic generated at the output of burst assemblers. In addition, a rule of thumb was derived for the relation between the average input traffic rate and the parameters of the hybrid burst assemblers that simplify the design and set-up of burst assembly units in practical situations.
- Using the traffic models developed in Chapter 5, it was demonstrated that the performance of an OBS core node using traces is quite similar to the results obtained when the burst arrivals to the core node is modeled with the Poisson process. In fact, usage of the Poisson as the process of the burst arrival to the core node leads to an upper bound on the burst drop rate at that node.

- Virtual optical bus is presented as a novel architecture for packet-based transport of Internet traffic in Chapter 6. The VOB paradigm is based on establishing a coordination among a set of ingress nodes in a network in a way that no collision occurs among packets belonging to these nodes. We formulated design of a VOB-based network, which consists of grouping all aggregate flows into VOBs as an ILP formulation with the objective of minimizing inter-VOB collisions. Through several design examples, the merits of the VOB network architecture are quantified. Our numerical results demonstrate that this approach can greatly alleviate the contention problem, which is central to the design of OBS networks.

9.2 Future Research

In the following, we outline some of the issues that can be considered as future studies in the area of the VOB networks.

- The VOB architecture is based on coordination among different sources in the network. This necessitates developments of a signaling scheme that takes care of setting up and tearing down VOBs inside the network. One possible approach for this purpose would be to design a suite of signaling messages to be incorporated into GMPLS control plane.

- In Section 8.3 we studied the impact of load variations on the performance of the VOB network. It is demonstrated that there is no need to redesign the layout of the network for short term variation of traffic. However, when the traffic variations are large, the designed VOB layout has to be adapted accordingly in order to achieve the desired performance in the network. This raises a couple of interesting research issues. For instance, an algorithm can be developed that takes care of adapting the VOB layout to the medium-term changes of traffic without requiring an abrupt change in the whole layout. Also, a control metric can be developed that would trigger redesigning of the VOB layout based on a new traffic matrix over the longer time periods.

- It is important to distinguish between different types of traffic flows and appropriately provide them with required services within the VOB network. In the simplest case, one should distinguish two types of traffic, namely streaming (e.g. voice over IP) and elastic traffic (e.g. file transfer). While the streaming traffic requires small delay and might tolerate some losses, the elastic traffic needs high throughput and can tolerate some delays. One interesting research topic would be to consider VOB layout design subject to QoS requirements.

Appendix A

Results of Solving ILP for Design Examples

Table A.1: Results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.6.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(0,5),(0,6),(0,7),(6,3),(7,4),(9,6)$
2	$(1 \rightarrow 0)$	$(1,0)$
3	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6)$	$(1,4),(1,5),(1,6),(2,6),(4,5),(4,6)$
4	$(1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(0,8),(1,8)$
5	$(2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$	$(2,7),(3,6),(3,7),(4,7)$
6	$(3 \rightarrow 4 \rightarrow 5)$	$(3,5)$
7	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7)$	$(1,7),(2,8),(3,2),(3,8),(8,7),(9,8)$
8	$(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2)$	$(1,2),(4,8),(4,9),(5,9),(8,0),(9,1)$
9	$(4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9)$	$(1,9),(2,1),(3,9),(4,0),(4,2)$
10	$(5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(0,1),(5,0),(5,7),(5,8),(7,0)$
11	$(5 \rightarrow 6)$	$(5,6)$
12	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5)$	$(2,4),(2,5),(6,0),(6,7),(6,9),(7,9),(9,2),(9,4)$
13	$(6 \rightarrow 7 \rightarrow 8)$	$(6,8),(7,8)$
14	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(1,3),(7,1),(8,1),(8,2),(9,3)$

Continued on next page

Table A.1 – continued from previous page

VOB id	Route	Associated O-D Flows
15	$(7 \rightarrow 6)$	$(7,6)$
16	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1)$	$(4,1),(5,1),(6,2),(6,5),$ $(7,2),(8,4),(8,5)$
17	$(8 \rightarrow 9)$	$(8,9)$
18	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9)$	$(0,9),(2,0),(2,9),(3,0),$ $(5,2),(5,4),(7,5),(8,3)$
19	$(9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1)$	$(3,1),(4,3),(5,3),(6,1),$ $(6,4),(7,3),(8,6),(9,5),$ $(9,7)$
20	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4)$	$(0,2),(0,3),(0,4),(2,3),$ $(3,4),(9,0)$

Table A.2: results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.7.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(0,6),(0,9),(7,6),(9,7),$ $(9,8)$
2	$(0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(0,1),(0,2),(0,4),(0,7),$ $(1,6),(2,6),(4,8),(6,8),$ $(7,8)$
3	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0)$	$(1,2),(1,4),(2,5),(3,7),$ $(3,8),(4,5),(5,8),(6,9),$ $(7,9),(8,0),(9,0)$
4	$(1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2)$	$(0,5),(1,7),(1,9),(4,2),$ $(6,2),(7,2),(7,4),(8,2),$ $(8,4),(9,4),(9,6)$
5	$(2 \rightarrow 1)$	(2,1)
6	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(3,4),(3,5),(4,1),(4,9),$ $(5,0),(6,1),(9,1)$
7	$(5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(0,8),(1,8),(2,8),(5,1),$ $(5,2)$
8	$(6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9)$	$(2,0),(2,9),(3,2),(3,9),$ $(4,0),(4,3),(5,9),(6,3),$ $(6,4)$
9	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(0,3),(1,3),(7,0),(7,1),$ $(8,1)$

Continued on next page

Table A.2 – continued from previous page

VOB id	Route	Associated O-D Flows
10	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0)$	$(1,0), (3,0), (3,1), (5,3), (5,4), (6,0), (7,3), (8,3), (8,6)$
11	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5)$	$(6,5), (7,5), (8,5), (8,7)$
12	$(8 \rightarrow 9)$	(8,9)
13	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$	$(1,5), (2,3), (2,4), (2,7), (3,6), (4,6), (4,7), (5,6), (5,7), (6,7), (9,2), (9,3), (9,5)$

Table A.3: results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.75.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4)$	$(0,8), (6,4), (7,5), (8,5), (9,4), (9,5)$
2	$(0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5)$	$(0,1), (0,4), (1,2), (1,5), (2,5), (4,5)$
3	$(1 \rightarrow 0 \rightarrow 9)$	$(0,9), (1,0), (1,9)$
4	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(1,4), (2,3), (2,4), (3,6), (3,9), (4,9), (5,9), (6,9)$
5	$(2 \rightarrow 1 \rightarrow 0)$	(2,0)
6	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7)$	$(0,7), (2,8), (2,9), (3,0), (3,2), (8,7), (9,7)$
7	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(3,4), (3,5), (3,8), (4,7), (4,8), (5,7), (5,8), (7,8)$
8	$(4 \rightarrow 3 \rightarrow 2 \rightarrow 1)$	$(2,1), (3,1), (4,2)$
9	$(5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(0,6), (1,6), (1,8), (4,0), (4,1), (4,6), (5,0), (5,1), (5,4), (8,6), (9,8)$
10	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(0,3), (1,3), (6,0), (6,1), (6,2), (6,8), (7,9), (8,2), (9,3)$
11	$(6 \rightarrow 5)$	$(6,5)$
12	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(7,0), (7,1), (8,1)$
13	$(7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2)$	$(5,2), (7,2), (7,3)$
14	$(8 \rightarrow 9 \rightarrow 0)$	$(8,0), (8,9), (9,0)$

Continued on next page

Table A.3 – continued from previous page

VOB id	Route	Associated O-D Flows
15	$(9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(4,3),(5,3),(6,3),(7,4)$ $(7,6),(8,3),(8,4),(9,6)$
16	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$	$(0,2),(0,5),(1,7),(2,6),$ $(2,7),(3,7),(5,6),(6,7),$ $(9,1),(9,2)$

Table A.4: Results obtained from solving the ILP for the ring network with uniform traffic matrix at A_{max}=0.7.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5)$	$(0,4),(0,5),(2,3),(3,5),(4,5)$
2	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5)$	$(1,2),(1,5),(2,4),(2,5)$
3	$(1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(0,6),(0,7),(1,0),(1,7)$
4	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6)$	$(1,3),(1,4),(1,6),(3,6),(5,6)$
5	$(2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$	$(2,6),(2,7),(3,7)$
6	$(2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(0,8),(1,8),(2,9)$
7	$(3 \rightarrow 4)$	$(3,4)$
8	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(1,9),(2,8),(3,1),(3,8)$
9	$(4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0)$	$(3,0),(4,0),(4,3)$
10	$(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(4,6),(4,7),(4,8),(6,8),(7,8)$
11	$(4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9)$	$(0,9),(2,0),(3,9),(4,2),(4,9)$
12	$(5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0)$	$(5,0),(5,7),(6,0),(7,9),(9,0)$
13	$(5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1)$	$(2,1),(4,1),(5,1)$
14	$(5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(5,8),(5,9),(6,9)$
15	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(6,1),(6,7),(7,1),(8,9),(9,1)$
16	$(6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2)$	$(3,2),(5,2),(5,3),(6,2)$
17	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2)$	$(0,2),(7,0),(7,2),(8,2)$
18	$(8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(0,1),(8,0),(8,1)$
19	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(6,3),(7,3),(8,3),(8,6),(8,7)$
20	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4)$	$(7,4),(8,4),(8,5)$
21	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(0,3),(9,2),(9,3)$
22	$(9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4)$	$(5,4),(6,4),(7,5),(7,6),(9,4),(9,7),(9,8)$
23	$(9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5)$	$(6,5),(9,5),(9,6)$

Table A.5: Results obtained from solving the ILP for the ring network with uniform traffic matrix at A_{max}=0.75.

VOB id	**Route**	**Associated O-D Flows**
1	$(1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4)$	$(1,0),(1,7),(1,8),(1,9),$ $(5,4),(6,4),(7,6),(8,4),$ $(9,4),(9,5)$
2	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(1,3),(1,4),(1,5),(2,6),$ $(3,7),(5,8),(5,9),(6,9),$ $(7,9)$
3	$(3 \rightarrow 4)$	$(3,4)$
4	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 6)$	$(3,5),(3,6)$
5	$(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0)$	$(4,5),(4,8),(4,9),(5,0),$ $(6,0),(8,0)$
6	$(4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(0,6),(1,6),(2,1),(2,8),$ $(3,0),(3,2),(4,0),(4,2),$ $(4,3),(8,6),(9,6)$
7	$(5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1)$	$(3,1),(4,1),(5,1),(5,2),$ $(5,3)$
8	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(1,2),(6,7),(7,0),(8,1),$ $(8,2),(8,3)$
9	$(6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7)$	$(0,7),(2,7),(2,9),(3,9),$ $(6,1),(6,2),(6,3),(6,5),$ $(8,7),(9,7),(9,8)$
10	$(7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(0,8),(0,9),(2,0),(3,8)$ $(7,1),(7,2),(7,3),(7,4)$
11	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5)$	$(7,5),(8,5)$
12	$(8 \rightarrow 9)$	$(8,9)$
13	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(0,3),(2,3),(9,1),(9,2),$ $(9,3)$
14	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(0,1),(0,2),(0,4),(0,5),$ $(2,4),(4,7),(5,6),(2,5),$ $(4,6),(5,7),(6,8),(7,8),$ $(9,0)$

Table A.6: Results obtained from solving the ILP for NSFNET with k=1.

VOB id	**Route**	**Associated O-D Flows**
1	$(0 \rightarrow 7 \rightarrow 6 \rightarrow 4)$	$(0,4),(0,6),(0,7),(6,4),(7,4),(7,6)$
2	$(0 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(0,8),(0,9),(7,8),(7,9),(8,9)$
3	$(0 \rightarrow 1 \rightarrow 3 \rightarrow 10)$	$(0,1),(0,3),(0,10),(1,3),(1,10),(3,10)$
4	$(0 \rightarrow 7 \rightarrow 8 \rightarrow 11)$	$(0,11),(7,11),(8,11)$
5	$(0 \rightarrow 2 \rightarrow 5 \rightarrow 12)$	$(0,2),(0,5),(0,12),(2,5),(2,12),(5,12)$
6	$(0 \rightarrow 7 \rightarrow 8 \rightarrow 13)$	$(0,13),(7,13),(8,13)$
7	$(1 \rightarrow 3 \rightarrow 4)$	$(1,4),(3,4)$
8	$(1 \rightarrow 0 \rightarrow 7 \rightarrow 6)$	$(1,6),(1,7)$
9	$(1 \rightarrow 0 \rightarrow 7 \rightarrow 8)$	$(1,8)$
10	$(1 \rightarrow 2 \rightarrow 5 \rightarrow 9)$	$(1,2),(1,5),(1,9),(2,9),(5,9)$
11	$(1 \rightarrow 3 \rightarrow 10 \rightarrow 11)$	$(1,11),(3,11),(10,11)$
12	$(1 \rightarrow 2 \rightarrow 5 \rightarrow 12)$	$(1,12)$
13	$(1 \rightarrow 3 \rightarrow 10 \rightarrow 13)$	$(1,13),(3,13),(10,13)$
14	$(2 \rightarrow 5 \rightarrow 4)$	$(2,4),(5,4)$
15	$(2 \rightarrow 0 \rightarrow 7 \rightarrow 6)$	$(2,0),(2,6),(2,7)$
16	$(2 \rightarrow 0 \rightarrow 7 \rightarrow 8)$	$(2,8)$
17	$(2 \rightarrow 1 \rightarrow 3 \rightarrow 10)$	$(2,1),(2,3),(2,10)$
18	$(2 \rightarrow 5 \rightarrow 12 \rightarrow 11)$	$(2,11),(5,11),(12,11)$
19	$(2 \rightarrow 5 \rightarrow 12 \rightarrow 13)$	$(2,13),(5,13),(12,13)$
20	$(3 \rightarrow 4 \rightarrow 6 \rightarrow 7)$	$(3,6),(3,7),(4,6),(4,7),(6,7)$
21	$(3 \rightarrow 10 \rightarrow 13 \rightarrow 8)$	$(3,8),(10,8),(13,8)$
22	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 9)$	$(3,5),(3,9),(4,5),(4,9)$
23	$(3 \rightarrow 10 \rightarrow 13 \rightarrow 12)$	$(3,12),(10,12),(13,12)$
24	$(4 \rightarrow 3 \rightarrow 1 \rightarrow 0)$	$(1,0),(3,0),(3,1),(4,0),(4,3)$
25	$(4 \rightarrow 3 \rightarrow 1)$	$(4,1)$
26	$(4 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(4,8),(6,8)$
27	$(4 \rightarrow 3 \rightarrow 10 \rightarrow 11)$	$(4,10),(4,11)$
28	$(4 \rightarrow 3 \rightarrow 10 \rightarrow 13)$	$(4,13)$
29	$(5 \rightarrow 4 \rightarrow 6 \rightarrow 7)$	$(5,6),(5,7)$
30	$(5 \rightarrow 9 \rightarrow 8)$	$(5,8),(9,8)$
31	$(5 \rightarrow 12 \rightarrow 13 \rightarrow 10)$	$(5,10),(12,10),(13,10)$
32	$(6 \rightarrow 7 \rightarrow 0 \rightarrow 1)$	$(6,0),(6,1),(7,0)$
33	$(6 \rightarrow 4 \rightarrow 5 \rightarrow 2)$	$(4,2),(5,2),(6,2),(6,5)$
34	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(6,9)$
35	$(6 \rightarrow 4 \rightarrow 3 \rightarrow 10)$	$(6,3),(6,10)$
36	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 11)$	$(6,11)$
37	$(6 \rightarrow 4 \rightarrow 5 \rightarrow 12)$	$(4,12),(6,12)$

Continued on next page

Table A.6 – continued from previous page

VOB id	Route	Associated O-D Flows
38	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 13)$	$(6,13)$
39	$(7 \rightarrow 0 \rightarrow 1)$	$(7,1)$
40	$(7 \rightarrow 0 \rightarrow 2)$	$(7,2)$
41	$(7 \rightarrow 0 \rightarrow 1 \rightarrow 3)$	$(7,3)$
42	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 5)$	$(7,5),(8,5),(9,5)$
43	$(7 \rightarrow 8 \rightarrow 13 \rightarrow 10)$	$(7,10)$
44	$(7 \rightarrow 8 \rightarrow 13 \rightarrow 12)$	$(7,12),(8,12)$
45	$(8 \rightarrow 7 \rightarrow 0 \rightarrow 1)$	$(8,0),(8,1),(8,7)$
46	$(8 \rightarrow 7 \rightarrow 0 \rightarrow 2)$	$(8,2)$
47	$(8 \rightarrow 13 \rightarrow 10 \rightarrow 3)$	$(8,3),(8,10),(10,3),(13,3)$
48	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 4)$	$(8,4),(8,6)$
49	$(9 \rightarrow 8 \rightarrow 7 \rightarrow 0)$	$(9,0),(9,7)$
50	$(9 \rightarrow 5 \rightarrow 2 \rightarrow 1)$	$(5,1),(9,1),(9,2)$
51	$(9 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(5,3),(9,3),(9,4)$
52	$(9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(9,6)$
53	$(9 \rightarrow 8 \rightarrow 13 \rightarrow 10)$	$(9,10),(9,13)$
54	$(9 \rightarrow 8 \rightarrow 11)$	$(9,11)$
55	$(9 \rightarrow 5 \rightarrow 12)$	$(9,12)$
56	$(10 \rightarrow 3 \rightarrow 1 \rightarrow 0)$	$(10,0),(10,1)$
57	$(10 \rightarrow 3 \rightarrow 1 \rightarrow 2)$	$(3,2),(10,2)$
58	$(10 \rightarrow 11 \rightarrow 12 \rightarrow 5)$	$(10,5),(11,5),(11,12),(12,5)$
59	$(10 \rightarrow 3 \rightarrow 4 \rightarrow 6)$	$(10,4),(10,6)$
60	$(10 \rightarrow 13 \rightarrow 8 \rightarrow 7)$	$(10,7),(13,7)$
61	$(10 \rightarrow 13 \rightarrow 8 \rightarrow 9)$	$(10,9),(13,9)$
62	$(11 \rightarrow 8 \rightarrow 7 \rightarrow 0)$	$(11,0),(11,7),(11,8)$
63	$(11 \rightarrow 10 \rightarrow 3 \rightarrow 1)$	$(11,1),(11,3)$
64	$(11 \rightarrow 12 \rightarrow 5 \rightarrow 2)$	$(11,2),(12,2)$
65	$(11 \rightarrow 10 \rightarrow 3 \rightarrow 4)$	$(11,4)$
66	$(11 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(11,6)$
67	$(11 \rightarrow 8 \rightarrow 9)$	$(11,9)$
68	$(11 \rightarrow 10)$	$(11,10)$
69	$(11 \rightarrow 12 \rightarrow 13)$	$(11,13)$
70	$(12 \rightarrow 5 \rightarrow 2 \rightarrow 0)$	$(5,0),(12,0)$
71	$(12 \rightarrow 5 \rightarrow 2 \rightarrow 1)$	$(12,1)$
72	$(12 \rightarrow 13 \rightarrow 10 \rightarrow 3)$	$(12,3)$
73	$(12 \rightarrow 5 \rightarrow 4 \rightarrow 6)$	$(12,4),(12,6)$
74	$(12 \rightarrow 13 \rightarrow 8 \rightarrow 7)$	$(12,7),(12,8)$
75	$(12 \rightarrow 5 \rightarrow 9)$	$(12,9)$
76	$(13 \rightarrow 8 \rightarrow 7 \rightarrow 0)$	$(13,0)$

Continued on next page

Table A.6 – continued from previous page

VOB id	Route	Associated O-D Flows
77	$(13 \rightarrow 10 \rightarrow 3 \rightarrow 1)$	$(13, 1)$
78	$(13 \rightarrow 12 \rightarrow 5 \rightarrow 2)$	$(13, 2), (13, 5)$
79	$(13 \rightarrow 10 \rightarrow 3 \rightarrow 4)$	$(13, 4)$
80	$(13 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(13, 6)$
81	$(13 \rightarrow 12 \rightarrow 11)$	$(13, 11)$

Table A.7: Results obtained from solving the ILP for NSFNET with k=2.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7)$	$(0, 4), (1, 4), (1, 6), (3, 7), (4, 7), (6, 7)$
2	$(0 \rightarrow 2 \rightarrow 5 \rightarrow 9 \rightarrow 8)$	$(0, 5), (0, 9), (2, 8), (2, 9), (5, 8),$ $(5, 9), (9, 8)$
3	$(1 \rightarrow 2)$	$(1, 2)$
4	$(1 \rightarrow 0 \rightarrow 7 \rightarrow 8 \rightarrow 13 \rightarrow 10)$	$(0, 8), (0, 13), (1, 7), (1, 8),$ $(1, 13), (7, 8), (7, 10), (7, 13),$ $(8, 13), (13, 10)$
5	$(1 \rightarrow 0 \rightarrow 7 \rightarrow 8 \rightarrow 11)$	$(0, 11), (1, 11), (7, 11)$
6	$(1 \rightarrow 0 \rightarrow 2 \rightarrow 5 \rightarrow 12)$	$(0, 2), (0, 12), (1, 12), (2, 12)$
7	$(2 \rightarrow 1)$	$(2, 1)$
8	$(2 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5)$	$(1, 3), (1, 5), (2, 4), (3, 5), (4, 5)$
9	$(2 \rightarrow 0 \rightarrow 7)$	$(2, 7)$
10	$(2 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow 10)$	$(0, 3), (0, 10), (1, 10), (2, 0), (2, 10)$
11	$(2 \rightarrow 5 \rightarrow 9 \rightarrow 8 \rightarrow 11)$	$(2, 11), (5, 11), (8, 11), (9, 11)$
12	$(2 \rightarrow 5 \rightarrow 9 \rightarrow 8 \rightarrow 13 \rightarrow 12)$	$(2, 13), (5, 13), (8, 12), (9, 12),$ $(9, 13), (13, 12)$
13	$(3 \rightarrow 10 \rightarrow 13 \rightarrow 8)$	$(3, 8), (10, 8), (13, 8)$
14	$(3 \rightarrow 1 \rightarrow 2 \rightarrow 5 \rightarrow 9)$	$(1, 9), (2, 5), (3, 9)$
15	$(3 \rightarrow 10 \rightarrow 11 \rightarrow 12 \rightarrow 13)$	$(3, 10), (3, 12), (3, 13), (10, 11), (11, 13)$
16	$(4 \rightarrow 5 \rightarrow 2 \rightarrow 0 \rightarrow 7 \rightarrow 6)$	$(0, 6), (0, 7), (2, 6), (4, 2),$ $(4, 6), (5, 6), (5, 7)$
17	$(4 \rightarrow 5 \rightarrow 12 \rightarrow 13 \rightarrow 10)$	$(4, 12), (4, 13), (5, 10), (12, 10)$
18	$(5 \rightarrow 12)$	$(5, 12)$
19	$(6 \rightarrow 7 \rightarrow 0 \rightarrow 2)$	$(6, 2), (7, 0)$
20	$(6 \rightarrow 4 \rightarrow 3 \rightarrow 1 \rightarrow 0)$	$(1, 0), (4, 0), (6, 0), (6, 1)$
21	$(6 \rightarrow 4 \rightarrow 3 \rightarrow 10 \rightarrow 11)$	$(3, 11), (4, 11), (6, 11)$
22	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 13 \rightarrow 12)$	$(6, 12), (6, 13)$

Continued on next page

Table A.7 – continued from previous page

VOB id	Route	Associated O-D Flows
23	$(6 \rightarrow 4 \rightarrow 3 \rightarrow 10)$	$(4,3),(4,10),(6,10)$
24	$(7 \rightarrow 0 \rightarrow 1)$	$(7,1)$
25	$(7 \rightarrow 6 \rightarrow 4 \rightarrow 3)$	$(6,3),(6,4),(7,3),(7,4)$
26	$(7 \rightarrow 6 \rightarrow 4 \rightarrow 5 \rightarrow 9 \rightarrow 8)$	$(4,8),(4,9),(6,5),(6,8),$ $(6,9),(7,5),(7,6),(7,9)$
27	$(7 \rightarrow 8 \rightarrow 13 \rightarrow 12)$	$(7,12)$
28	$(8 \rightarrow 9 \rightarrow 5 \rightarrow 4 \rightarrow 6)$	$(8,4),(8,5),(9,4),(9,6)$
29	$(9 \rightarrow 5 \rightarrow 2 \rightarrow 0 \rightarrow 1)$	$(5,0),(9,0),(9,1),(9,2),(9,5)$
30	$(9 \rightarrow 5 \rightarrow 2 \rightarrow 1 \rightarrow 3)$	$(2,3),(9,3)$
31	$(9 \rightarrow 5 \rightarrow 4 \rightarrow 6 \rightarrow 7)$	$(9,7)$
32	$(9 \rightarrow 8 \rightarrow 11 \rightarrow 10)$	$(8,10),(9,10)$
33	$(10 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 0)$	$(3,0),(3,2),(10,0),(10,2)$
34	$(10 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 2 \rightarrow 1)$	$(3,4),(4,1),(5,1),(5,2),(10,4),$ $(10,5)$
35	$(10 \rightarrow 13 \rightarrow 12 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(5,3),(5,4),(10,3),(10,13),(12,3),$ $(12,4),(13,3),(13,4),(13,5)$
36	$(10 \rightarrow 11 \rightarrow 12 \rightarrow 5 \rightarrow 4)$	$(10,12),(11,4)$
37	$(10 \rightarrow 11 \rightarrow 8 \rightarrow 7)$	$(10,7),(11,7)$
38	$(10 \rightarrow 11 \rightarrow 8 \rightarrow 9)$	$(10,9),(11,8)$
39	$(11 \rightarrow 10 \rightarrow 3 \rightarrow 1)$	$(3,1),(10,1),(11,1)$
40	$(11 \rightarrow 12 \rightarrow 5)$	$(11,5),(11,12),(12,5)$
41	$(11 \rightarrow 12 \rightarrow 13)$	$(12,13)$
42	$(12 \rightarrow 5 \rightarrow 2 \rightarrow 1)$	$(12,1)$
43	$(12 \rightarrow 11 \rightarrow 8 \rightarrow 7 \rightarrow 0 \rightarrow 2)$	$(8,2),(11,0),(11,2),(12,0),(12,2)$
44	$(12 \rightarrow 11 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(8,6),(8,7),(11,6),(12,6),$ $(12,7),(12,8)$
45	$(12 \rightarrow 11 \rightarrow 8 \rightarrow 9)$	$(11,9),(12,9),(12,11)$
46	$(13 \rightarrow 8 \rightarrow 7 \rightarrow 0 \rightarrow 1)$	$(0,1),(8,0),(8,1),(13,0),(13,1)$
47	$(13 \rightarrow 8 \rightarrow 7 \rightarrow 0 \rightarrow 2)$	$(7,2),(13,2),(13,7)$
48	$(13 \rightarrow 8 \rightarrow 11 \rightarrow 10 \rightarrow 3)$	$(8,3),(11,3),(11,10),(13,11)$
49	$(13 \rightarrow 10 \rightarrow 3 \rightarrow 4 \rightarrow 6)$	$(3,6),(10,6),(13,6)$
50	$(13 \rightarrow 8 \rightarrow 9)$	$(8,9),(13,9)$

Table A.8: Results obtained from solving the ILP for NSFNET with k=3.

VOB id	**Route**	**Associated O-D Flows**
1	$(0 \rightarrow 2 \rightarrow 5 \rightarrow 4 \rightarrow 6)$	$(0,2),(2,4),(2,6)$
2	$(0 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7)$	$(0,1),(0,3),(0,4),(0,6),$ $(1,3),(1,6),(3,7),(4,7)$
3	$(0 \rightarrow 1 \rightarrow 3 \rightarrow 10 \rightarrow 11)$	$(0,10),(1,10),(1,11),(3,10)$
4	$(1 \rightarrow 0 \rightarrow 7 \rightarrow 8 \rightarrow 13 \rightarrow 10)$	$(1,7),(1,8),(1,13),(7,8),$ $(7,10),(7,13),(13,10)$
5	$(1 \rightarrow 2 \rightarrow 5 \rightarrow 12 \rightarrow 13 \rightarrow 10)$	$(1,5),(1,12),(2,5),(2,10),(5,12),$ $(5,13),(12,10)$
6	$(2 \rightarrow 1 \rightarrow 0)$	$(1,0),(2,0),(2,1)$
7	$(2 \rightarrow 1 \rightarrow 3)$	$(2,3)$
8	$(2 \rightarrow 0 \rightarrow 7 \rightarrow 8 \rightarrow 11)$	$(0,7),(0,11),(2,11),(7,11)$
9	$(2 \rightarrow 0 \rightarrow 7 \rightarrow 8 \rightarrow 13 \rightarrow 12)$	$(0,8),(0,12),(0,13),(2,8),$ $(2,12),(2,13),(7,12),(8,12)$
10	$(3 \rightarrow 1 \rightarrow 2 \rightarrow 5 \rightarrow 9)$	$(1,9),(2,9),(3,2),(3,5)$
11	$(3 \rightarrow 10 \rightarrow 13 \rightarrow 8 \rightarrow 9)$	$(3,8),(3,9),(10,9)$
12	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 12 \rightarrow 11 \rightarrow 10)$	$(3,11),(3,12),(4,11),(5,10),$ $(5,11),(11,10)$
13	$(4 \rightarrow 5 \rightarrow 2 \rightarrow 0)$	$(4,0),(4,2),(4,5)$
14	$(4 \rightarrow 5 \rightarrow 9 \rightarrow 8)$	$(4,8),(4,9),(5,8)$
15	$(6 \rightarrow 7 \rightarrow 0 \rightarrow 2 \rightarrow 5 \rightarrow 9)$	$(0,5),(0,9),(5,9),(6,0),(6,2),(6,5),$ $(6,9),(7,0),(7,5),(7,9)$
16	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 11 \rightarrow 12)$	$(6,7),(6,8),(6,11),(11,12)$
17	$(6 \rightarrow 4 \rightarrow 3 \rightarrow 10 \rightarrow 13)$	$(3,13),(4,3),(4,10),(6,3),$ $(6,10),(10,13)$
18	$(6 \rightarrow 4 \rightarrow 5 \rightarrow 12 \rightarrow 13)$	$(4,12),(4,13),(6,4),(6,12),$ $(6,13),(12,13)$
19	$(7 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow 4)$	$(1,4),(7,1)$
20	$(7 \rightarrow 6 \rightarrow 4 \rightarrow 3 \rightarrow 1)$	$(3,1),(4,1),(6,1),(7,3),(7,4)$
21	$(8 \rightarrow 13 \rightarrow 10 \rightarrow 3 \rightarrow 1 \rightarrow 0)$	$(3,0),(8,0),(8,1),(8,13),(10,0),$ $(13,0),(13,1)$
22	$(8 \rightarrow 11 \rightarrow 10 \rightarrow 3 \rightarrow 4 \rightarrow 6)$	$(3,4),(3,6),(4,6),(8,4),(8,10),$ $(11,3),(11,4),(11,6)$
23	$(9 \rightarrow 5 \rightarrow 2 \rightarrow 1 \rightarrow 0)$	$(5,1),(9,0),(9,1)$
24	$(9 \rightarrow 8 \rightarrow 13 \rightarrow 10 \rightarrow 3)$	$(8,3),(9,3),(9,10),(9,13),$ $(10,3),(13,3)$
25	$(9 \rightarrow 5 \rightarrow 4 \rightarrow 6)$	$(5,6),(9,4),(9,5),(9,6)$
26	$(9 \rightarrow 5 \rightarrow 2 \rightarrow 0 \rightarrow 7)$	$(2,7),(5,0),(5,7),(9,2),(9,7)$

Continued on next page

Table A.8 – continued from previous page

VOB id	Route	Associated O-D Flows
27	$(9 \rightarrow 8)$	$(9,8)$
28	$(9 \rightarrow 8 \rightarrow 11 \rightarrow 12)$	$(8,11),(9,11),(9,12)$
29	$(10 \rightarrow 13 \rightarrow 12 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(5,3),(5,4),(10,5),(13,4)$
30	$(10 \rightarrow 11 \rightarrow 12 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(10,4),(10,11),(10,12),(12,3),$ $(12,4),(12,5)$
31	$(10 \rightarrow 11 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(8,6),(8,7),(10,6),(10,7),$ $(10,8),(11,7)$
32	$(11 \rightarrow 12 \rightarrow 5 \rightarrow 2 \rightarrow 1)$	$(5,2),(11,5),(12,1)$
33	$(11 \rightarrow 10 \rightarrow 3 \rightarrow 1 \rightarrow 2)$	$(1,2),(10,1),(10,2),(11,1)$
34	$(11 \rightarrow 10 \rightarrow 13)$	$(11,13)$
35	$(12 \rightarrow 11 \rightarrow 8 \rightarrow 7 \rightarrow 0 \rightarrow 2)$	$(8,2),(11,0),(11,2),(12,0),$ $(12,2),(12,8)$
36	$(12 \rightarrow 13 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(7,6),(12,6),(12,7),(13,6)$
37	$(12 \rightarrow 11 \rightarrow 8 \rightarrow 9)$	$(8,9),(12,9),(12,11)$
38	$(13 \rightarrow 8 \rightarrow 7 \rightarrow 0 \rightarrow 2)$	$(7,2),(13,2),(13,7)$
39	$(13 \rightarrow 8 \rightarrow 9 \rightarrow 5)$	$(8,5),(13,5),(13,8),(13,9)$
40	$(13 \rightarrow 12 \rightarrow 11 \rightarrow 8 \rightarrow 9)$	$(11,8),(11,9),(13,11),(13,12)$

Table A.9: Results obtained from solving the ILP for NSFNET with k=4.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 1)$	$(0,1)$
2	$(0 \rightarrow 2 \rightarrow 5 \rightarrow 12 \rightarrow 13 \rightarrow 8)$	$(0,8),(0,12),(0,13),(2,8),$ $(2,12),(2,13),(5,12),(5,13),$ $(12,8),(12,13),(13,8)$
3	$(0 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(0,7),(0,9),(7,9)$
4	$(1 \rightarrow 0 \rightarrow 7 \rightarrow 6 \rightarrow 4 \rightarrow 3)$	$(0,3),(0,4),(0,6),(1,4),$ $(1,6),(6,3),(7,3)$
5	$(1 \rightarrow 0 \rightarrow 7 \rightarrow 8 \rightarrow 11 \rightarrow 10)$	$(0,10),(0,11),(1,7),(1,8),$ $(7,10),(7,11)$
6	$(1 \rightarrow 3 \rightarrow 10 \rightarrow 13 \rightarrow 12)$	$(1,3),(1,12),(1,13),$ $(3,10),(3,13)$
7	$(2 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow 10)$	$(2,10)$
8	$(2 \rightarrow 1 \rightarrow 3 \rightarrow 10 \rightarrow 11)$	$(1,10),(1,11),(2,11),(3,11)$
9	$(3 \rightarrow 10 \rightarrow 13 \rightarrow 8 \rightarrow 7)$	$(3,7),(3,8),(10,7)$
10	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 12 \rightarrow 13 \rightarrow 10)$	$(3,4),(3,12),(4,10),(4,12),$

Continued on next page

Table A.9 – continued from previous page

VOB id	Route	Associated O-D Flows
		$(4,13),(5,10),(12,10)$
11	$(4 \rightarrow 6)$	$(4,6)$
12	$(4 \rightarrow 5 \rightarrow 9 \rightarrow 8 \rightarrow 7)$	$(4,7),(4,8),(5,8),(8,7),(9,7)$
13	$(4 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 5 \rightarrow 9)$	$(1,9),(2,9),(3,1),(3,5),$ $(3,9),(4,1),(4,2),(4,9)$
14	$(5 \rightarrow 2 \rightarrow 0 \rightarrow 7 \rightarrow 6 \rightarrow 4)$	$(2,0),(2,4),(2,6),(2,7),$ $(5,4),(5,6),(5,7),(6,4)$
15	$(6 \rightarrow 7 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 5)$	$(0,2),(0,5),(1,2),(1,5),(2,5),$ $(6,0),(6,1),(6,2),(7,5)$
16	$(6 \rightarrow 7 \rightarrow 0 \rightarrow 2 \rightarrow 5 \rightarrow 9)$	$(5,9),(6,7),(6,9),(7,2)$
17	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 11 \rightarrow 10)$	$(6,8),(6,10)$
18	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 11 \rightarrow 12)$	$(6,11),(7,8),(7,12)$
19	$(7 \rightarrow 0 \rightarrow 2 \rightarrow 1)$	$(7,1)$
20	$(7 \rightarrow 6 \rightarrow 4 \rightarrow 5 \rightarrow 12 \rightarrow 13)$	$(4,5),(6,5),(6,12),(6,13),$ $(7,6),(7,13)$
21	$(8 \rightarrow 9 \rightarrow 5 \rightarrow 2 \rightarrow 1)$	$(2,1),(5,2),(8,1),(8,2),(8,5),$ $(8,9),(9,1),(9,2),(9,5)$
22	$(8 \rightarrow 9 \rightarrow 5 \rightarrow 4 \rightarrow 6)$	$(8,6),(9,4),(9,6)$
23	$(8 \rightarrow 11 \rightarrow 12 \rightarrow 5 \rightarrow 9)$	$(8,11),(8,12),(11,5),(11,12),$ $(12,5),(12,9)$
24	$(9 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 1 \rightarrow 0)$	$(4,0),(5,0),(5,1),(5,3),(9,0)$
25	$(9 \rightarrow 8 \rightarrow 13 \rightarrow 10 \rightarrow 3 \rightarrow 4 \rightarrow 6)$	$(8,3),(8,10),(8,13),(9,3),$ $(9,10),(9,13),(10,3),(10,6),$ $(13,6),(13,10)$
26	$(9 \rightarrow 8)$	$(9,8)$
27	$(9 \rightarrow 8 \rightarrow 13 \rightarrow 12 \rightarrow 11)$	$(9,11),(9,12),(12,11),(13,12)$
28	$(10 \rightarrow 13 \rightarrow 12 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(4,3),(10,5),(10,13),(12,3),$ $(12,4),(13,4),(13,5)$
29	$(10 \rightarrow 13 \rightarrow 12 \rightarrow 5 \rightarrow 2 \rightarrow 1 \rightarrow 3)$	$(2,3),(10,1),(10,2),(10,12),$ $(12,2),(13,1),(13,2),(13,3)$
30	$(10 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 12 \rightarrow 11)$	$(4,11),(5,11),(10,4)$
31	$(10 \rightarrow 11 \rightarrow 8 \rightarrow 13)$	$(10,11),(11,13)$
32	$(11 \rightarrow 10 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 0)$	$(1,0),(3,0),(3,2),(10,0),$ $(11,0),(11,1),(11,2),(11,3)$
33	$(11 \rightarrow 10 \rightarrow 3 \rightarrow 4 \rightarrow 6)$	$(3,6),(11,4),(11,6)$
34	$(11 \rightarrow 8 \rightarrow 13 \rightarrow 10)$	$(11,10)$
35	$(12 \rightarrow 13 \rightarrow 8 \rightarrow 7 \rightarrow 0)$	$(7,0),(8,0),(12,0),$ $(13,0),(13,7)$
36	$(12 \rightarrow 5 \rightarrow 2 \rightarrow 0 \rightarrow 1)$	$(12,1)$

Continued on next page

Table A.9 – continued from previous page

VOB id	Route	Associated O-D Flows
37	$(12 \rightarrow 11 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 4)$	$(7,4),(8,4),(11,7),(11,8),$ $(12,6),(12,7)$
38	$(13 \rightarrow 10 \rightarrow 11 \rightarrow 8 \rightarrow 9)$	$(10,8),(10,9),(11,9),$ $(13,9),(13,11)$

Table A.10: Results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.6 and -20% load change.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5)$	$(0,6),(0,8),(6,5),(7,5),$ $(9,7)$
2	$(0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(0,1),(0,3),(1,4),(1,7),$ $(2,3),(2,6),(3,8),(4,8),$ $(5,7),(5,8),(7,8)$
3	$(1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(0,7),(0,9),(1,0),(1,6),$ $(1,8),(7,6)$
4	$(2 \rightarrow 1 \rightarrow 0 \rightarrow 9)$	$(1,9),(2,0),(2,1),(2,9)$
5	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7)$	$(2,8),(3,0),(3,2),(3,7),$ $(3,9),(8,7)$
6	$(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0)$	$(4,0),(4,5),(4,6),(4,9),$ $(5,0),(5,9),(6,0),(9,0)$
7	$(5 \rightarrow 4 \rightarrow 3)$	$(4,3),(5,3)$
8	$(6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1)$	$(3,1),(5,1),(5,4),(6,2),$ $(6,3)$
9	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(6,8),(6,9),(7,9),(8,9)$
10	$(7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2)$	$(5,2),(7,2),(7,3)$
11	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6)$	$(0,5),(1,2),(2,5),(3,6),$ $(5,6),(7,0),(7,1),(8,1),$ $(9,3)$
12	$(8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4)$	$(0,4),(2,4),(3,4),(8,0),$ $(8,2),(8,3)$
13	$(9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1)$	$(4,1),(4,2),(6,1),(6,4),$ $(7,4),(8,4),(8,5),(8,6),$ $(9,4),(9,5),(9,6),(9,8)$
14	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$	$(0,2),(1,3),(1,5),(2,7),$ $(3,5),(4,7),(6,7),(9,1),$

Continued on next page

Table A.10 – continued from previous page

VOB id	Route	Associated O-D Flows
		$(9, 2)$

Table A.11: Results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.6 and +20% load change.

VOB id	Route	Associated O-D Flows
1	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(0, 7), (0, 9), (5, 4), (7, 3), (8, 3), (9, 6)$
2	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5)$	$(0, 5), (0, 8), (8, 5), (9, 5)$
3	$(1 \rightarrow 2)$	$(1, 2)$
4	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6)$	$(1, 4), (2, 4), (2, 6), (4, 6)$
5	$(2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(2, 5), (4, 0), (7, 1), (8, 1)$
6	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(1, 6), (1, 9), (3, 1), (9, 7)$
7	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(3, 5), (3, 8), (5, 7), (5, 8), (6, 8)$
8	$(3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9)$	$(3, 4), (3, 7), (4, 7), (4, 8), (4, 9)$
9	$(4 \rightarrow 5)$	$(4, 5)$
10	$(4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7)$	$(1, 7), (2, 7), (4, 1), (4, 2)$
11	$(5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(0, 3), (5, 6), (5, 9), (6, 0), (7, 9), (9, 2)$
12	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(0, 1), (1, 3), (2, 3), (6, 1), (6, 7), (7, 8), (8, 0)$
13	$(6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4)$	$(0, 2), (0, 4), (6, 9), (7, 0), (9, 4)$
14	$(6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(1, 8), (2, 8), (4, 3), (5, 2), (6, 2), (6, 4), (6, 5), (9, 8)$
15	$(6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9)$	$(2, 1), (2, 9), (3, 2), (3, 9), (5, 3), (6, 3)$
16	$(7 \rightarrow 6)$	$(7, 6)$
17	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0)$	$(1, 0), (2, 0), (5, 1), (7, 2), (8, 6)$
18	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0)$	$(3, 0), (5, 0), (7, 4), (7, 5), (8, 4), (8, 7)$
19	$(8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6)$	$(0, 6), (1, 5), (3, 6), (8, 2), (8, 9), (9, 0), (9, 3)$

Continued on next page

Table A.11 – continued from previous page

VOB id	Route	Associated O-D Flows
20	$(9 \rightarrow 0 \rightarrow 1)$	$(9,1)$

Table A.12: Results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.7 and -20% load change.

VOB id	Route	Associated O-D Flows
1	$(2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(0,7),(0,9),(1,0),(2,0),$ $(2,6),(2,8),(5,3),(6,3),$ $(7,3),(8,3),(8,6),(9,5),$ $(9,7)$
2	$(2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$	$(2,5),(2,7),(3,6),(3,7),$ $(5,7),(6,7)$
3	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(0,6),(1,6),(1,7),(3,0),$ $(3,1),(3,8),(3,9),(7,6),$ $(9,6),(9,8)$
4	$(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2)$	$(0,1),(0,2),(1,2),(4,5),$ $(4,6),(4,7),(4,8),(4,9),$ $(5,0),(6,8),(8,2),(8,9),$ $(9,1),(9,2)$
5	$(5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4)$	$(0,3),(0,4),(1,4),(2,3),$ $(3,4),(5,6),(5,8),(5,9),$ $(6,1),(6,9),(7,8),(8,0),$ $(8,1),(9,0)$
6	$(6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(0,8),(1,8),(1,9),(2,1),$ $(2,9),(3,2),(4,0),(4,1),$ $(4,3),(5,1),(6,0),(6,4),$ $(6,5)$
7	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5)$	$(0,5),(1,3),(1,5),(2,4),$ $(3,5),(7,0),(7,1),(7,9),$ $(9,3),(9,4)$
8	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2)$	$(4,2),(5,2),(5,4),(6,2),$ $(7,2),(7,4),(7,5),(8,4),$ $(8,5),(8,7)$

Table A.13: Results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.7 and -10% load change.

VOB id	**Route**	**Associated O-D Flows**
1	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3)$	$(0,6),(0,7),(0,9),(5,3),$ $(6,3),(6,4),(6,5),(8,6),$ $(9,4)$
2	$(0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4)$	$(0,1),(0,4),(1,4),(2,4)$
3	$(1 \rightarrow 2)$	$(1,2)$
4	$(2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(2,3),(2,5),(2,6),(3,6),$ $(3,8),(5,0),(5,7),(6,7),$ $(7,8),(7,9),(8,9),(9,1)$
5	$(2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(0,8),(2,0),(2,1),(2,8),$ $(2,9),(9,8)$
6	$(3 \rightarrow 4)$	$(3,4)$
7	$(3 \rightarrow 4 \rightarrow 5)$	$(3,5),(4,5)$
8	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7)$	$(1,8),(1,9),(2,7),(3,1),$ $(8,7),(9,7)$
9	$(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0)$	$(4,7),(4,9),(5,8),(6,0),$ $(6,8),(6,9),(8,0),(9,0)$
10	$(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(1,3),(4,1),(4,6),(4,8),$ $(5,9),(6,1),(6,2),(8,1),$ $(9,2),(9,3)$
11	$(4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5)$	$(0,5),(1,5),(1,6),(1,7),$ $(3,0),(3,7),(3,9),(4,2),$ $(8,5),(9,5),(9,6)$
12	$(5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0)$	$(1,0),(4,0),(5,1),(5,2),$ $(5,4)$
13	$(5 \rightarrow 6)$	$(5,6)$
14	$(7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3)$	$(0,2),(0,3),(7,0),(7,1),$ $(8,2)$
15	$(7 \rightarrow 6 \rightarrow 5)$	$(7,5),(7,6)$
16	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2)$	$(3,2),(4,3),(7,2),(7,3),$ $(7,4),(8,3),(8,4)$

Table A.14: Results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.7 and +10% load change.

VOB id	**Route**	**Associated O-D Flows**
1	$(0 \to 9 \to 8 \to 7 \to 6 \to 5 \to 4 \to 3 \to 2 \to 1)$	$(0,5),(0,6),(4,1),(5,1),$ $(6,1),(6,5),(8,4),(9,5),$ $(9,8)$
2	$(0 \to 1 \to 2 \to 3 \to 4 \to 5 \to 6 \to 7)$	$(0,1),(1,5),(1,6),(2,5),$ $(4,6),(5,7),(6,7)$
3	$(1 \to 2 \to 3 \to 4 \to 5 \to 6 \to 7 \to 8 \to 9 \to 0)$	$(1,2),(2,4),(2,6),(2,7),$ $(5,9),(7,0),(7,8)$
4	$(1 \to 0 \to 9 \to 8 \to 7)$	$(0,9),(1,0),(1,7),(1,8)$
5	$(3 \to 4)$	$(3,4)$
6	$(4 \to 5 \to 6 \to 7 \to 8 \to 9 \to 0 \to 1 \to 2 \to 3)$	$(4,5),(4,7),(4,9),(6,2),$ $(6,9),(8,3),(9,2),(9,3)$
7	$(5 \to 6)$	$(5,6)$
8	$(5 \to 4 \to 3 \to 2 \to 1 \to 0 \to 9 \to 8 \to 7 \to 6)$	$(0,7),(0,8),(2,8),(5,0),$ $(5,2),(7,6),(8,6),(8,7)$
9	$(6 \to 5 \to 4 \to 3 \to 2 \to 1 \to 0 \to 9)$	$(2,9),(3,2),(3,9),(6,0),$ $(6,3)$
10	$(7 \to 8 \to 9 \to 0 \to 1)$	$(7,1)$
11	$(7 \to 6 \to 5)$	$(7,5)$
12	$(7 \to 8 \to 9)$	$(7,9),(8,9)$
13	$(7 \to 6 \to 5 \to 4 \to 3 \to 2 \to 1 \to 0 \to 9)$	$(1,9),(3,0),(4,0),(4,2),$ $(5,4),(6,4),(7,3),(7,4)$
14	$(8 \to 9 \to 0 \to 1 \to 2 \to 3 \to 4 \to 5)$	$(1,3),(1,4),(3,5),(8,0),$ $(8,1),(9,1)$
15	$(9 \to 8 \to 7 \to 6 \to 5 \to 4 \to 3 \to 2 \to 1 \to 0)$	$(2,0),(2,1),(3,1),(4,3),$ $(5,3),(7,2),(8,2),(8,5),$ $(9,4),(9,6),(9,7)$
16	$(9 \to 0 \to 1 \to 2 \to 3 \to 4 \to 5 \to 6 \to 7 \to 8)$	$(0,2),(0,3),(0,4),(2,3),$ $(3,6),(3,7),(3,8),(4,8),$ $(5,8),(6,8),(9,0)$

Table A.15: Results obtained from solving the ILP for the ring network with random traffic matrix at A_{max}=0.7 and +20% load change.

VOB id	**Route**	**Associated O-D Flows**
1	$(0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5)$	$(0,5),(0,9),(7,5),(9,7)$
2	$(0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$	$(0,2),(0,3),(0,4),(3,6),$ $(3,8),(4,6),(4,8),(6,8)$
3	$(1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0)$	$(1,3),(1,6),(3,5),(6,9),$ $(7,0),(9,0)$
4	$(1 \rightarrow 2)$	$(1,2)$
5	$(2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(2,3),(2,4),(2,5),(5,1),$ $(5,6),(6,1)$
6	$(2 \rightarrow 1 \rightarrow 0 \rightarrow 9)$	$(1,9),(2,9)$
7	$(3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4)$	$(0,6),(0,7),(1,0),(3,1),$ $(3,9),(5,4),(6,4),(6,5),$ $(7,4),(9,4),(9,5),(9,8)$
8	$(4 \rightarrow 3)$	$(4,3)$
9	$(4 \rightarrow 5)$	$(4,5)$
10	$(5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 0 \rightarrow 1)$	$(5,7),(5,8),(6,0),(6,7),$ $(7,1),(8,1)$
11	$(5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6)$	$(0,8),(1,8),(2,0),(2,1),$ $(2,6),(4,0),(4,1),(4,2),$ $(5,9),(8,6)$
12	$(5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8)$	$(2,8),(4,9),(5,2),(5,3)$
13	$(7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2)$	$(3,2),(6,2),(6,3),(7,2),$ $(7,6)$
14	$(7 \rightarrow 8 \rightarrow 9)$	$(7,8),(7,9),(8,9)$
15	$(8 \rightarrow 7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0)$	$(3,0),(5,0),(7,3),(8,2),$ $(8,5)$
16	$(8 \rightarrow 9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4)$	$(0,1),(1,4),(3,4),(8,0),$ $(8,3),(8,4)$
17	$(8 \rightarrow 7)$	$(8,7)$
18	$(9 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$	$(1,5),(1,7),(2,7),(3,7),$ $(4,7),(9,1),(9,2),(9,3),$ $(9,6)$

List of Publications

Journals and Book Chapters

- A. Rostami and A. Wolisz, "Virtual optical bus: an efficient architecture for packet-based optical transport networks", *IEEE/OSA Journal of Optical Communications and Networking (JOCN)*, vol. 2, no. 11, pp. 901-914, 2010.

- A. Rostami, A. Wolisz, and A. Feldmann, "Traffic analysis in optical burst switching networks: a trace-based case study", *European Transactions on Telecommunications (ETT)*, vol. 20, no. 7, pp. 633-649, 2009, invited paper.

- J. A. Hernandez, V. Lopez, J. L. G. Dorado, R. Nejabati, H. Overby, A. Rostami, K. Vlachos, and G. Zervas, "Optical burst switching", In J. Aracil and F. Callegati, editors, *Enabling Optical Internet with Advanced Network Technologies*, chapter 4, Springer, London, 2009, ISBN: 978-1-84882-277-1.

- A. Rostami and A. Wolisz, "Modeling and Synthesis of Traffic in Optical Burst-Switched Networks", *IEEE/OSA Journal of Lightwave Technology (JLT)*, vol. 25, no. 10, pp. 2942-2952, 2007.

- A. Rostami and S. S. Chakraborty, "On Performance of Optical Buffers with Specific Number of Circulations", *IEEE Photonics Technology Letters*, vol. 17, no. 7, pp. 1570-1572, 2005.

Conference Proceedings

- A. Rostami and A. Wolisz, "Impact of edge traffic aggregation on the performance of FDL-assisted optical core switching nodes", In Proc. of *IEEE International Conference on Communications (ICC 2007)*, Glasgow, Scotland, June 2007.

- A. Rostami and A. Wolisz, "Modeling and Fast Emulation of Burst Traffic in Optical Burst-Switched Networks", In Proc. of *IEEE International Conference on Communications (ICC 2006)*, lstanbul, Turkey, June 2006.

- A. Rostami and A. Wolisz, "Impact of Burst Assembly on the Performance of FDL Buffers in Optical Burst-Switched Networks", In Proc. of *3rd International Symposium on Telecommunications (IST 2005)*, Shiraz, Iran, September 2005.

- A. Rostami and S. S. Chakraborty, "On Performance of Optical Buffers with Fixed Length Packets", In Proc. of *2nd IFIP WOCN Conference (WOCN 2005)*, Dubai, UAE, March 2005.

Technical Reports

- A. Rostami, "Virtual Optical Bus: A Novel Packet-Based Architecture for Optical Transport Networks", TKN Technical Report Series TKN-10-002, Telecommunication Networks Group, Technische Universität Berlin, February 2010.
- A. Rostami, "Modeling of Burst Assembly Process in Optical Burst-Switched Networks", Technical Report TKN-07-001, Telecommunication Networks Group, Technische Universität Berlin, February 2007.

Bibliography

[1] K. H. Liu, *IP Over Wdm*, Wiley, 2002.

[2] J. Postel et al., "Internet protocol", Internet Engineering Task Force (IETF), RFC 791, September 1981.

[3] D. Comer, *Internetworking with TCP/IP: principles, protocols, and architecture*, Prentice Hall, 2006.

[4] A. Odlyzko, "Minnesota Internet Traffic Studies (MINTS)", *University of Minnesota, www.dtc.umn.edu/mints/home.php*.

[5] ITU Rec. G.805, "Generic functional architecture of transport networks", International Telecommunication Union, ITU-T, March 2000.

[6] B. Mukherjee, *Optical WDM networks*, Springer-Verlag New York Inc., 2006.

[7] G. Bonaventura, G. Jones and S. Trowbridge, "Optical transport network evolution: hot standardization topics in ITU-T including standards coordination aspects", *IEEE Communications Magazine*, vol. 46, n. 10, pp. 124–131, 2008.

[8] I. Chlamtac, A. Ganz and G. Karmi, "Lightpath communications: An approach to high bandwidth optical WAN's", *IEEE Transactions on Communications*, vol. 40, n. 7, pp. 1171–1182, 1992.

[9] S. Baroni and P. Bayvel, "Wavelength requirements in arbitrarily connected wavelength-routed optical networks", *Journal of lightwave technology*, vol. 15, n. 2, pp. 242–251, 1997.

[10] D. Banerjee and B. Mukherjee, "Wavelength-routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study", *IEEE/ACM Transactions on Networking (TON)*, vol. 8, n. 5, pp. 598–607, 2000.

[11] R. Ramaswami and K.N. Sivarajan, *Optical networks: a practical perspective*, Morgan Kaufmann, 2002.

[12] L. Li and A.K. Somani, "Dynamic wavelength routing using congestion and neighborhood information", *IEEE/ACM Transactions on Networking (TON)*, vol. 7, n. 5, pp. 786, 1999.

[13] H. Zang, J. P. Jue, L. Sahasrabuddhe, R. Ramamurthy and B. Mukherjee, "Dynamic lightpath establishment in wavelength routed WDM networks", *IEEE Communications Magazine*, vol. 39, n. 9, pp. 100–108, 2001.

[14] A. Zapata-Beghelli and P. Bayvel, "Dynamic versus static wavelength-routed optical networks", *Journal of Lightwave Technology*, vol. 26, n. 20, pp. 3403–3415, 2008.

[15] C. Guillemot, M. Renaud, P. Gambini et al., "Transparent optical packet switching: The European ACTS KEOPS project approach", *Journal of lightwave technology*, vol. 16, n. 12, pp. 2117, 1998.

[16] M.J. O'Mahony, D. Simeonidou, D.K. Hunter and A. Tzanakaki, "The application of optical packet switching in future communication networks", *IEEE Communications Magazine*, vol. 39, n. 3, pp. 128–135, 2001.

[17] S. Yao, S.J.B. Yoo, B. Mukherjee and S. Dixit, "All-optical packet switching for metropolitan area networks: Opportunities and challenges", *IEEE Communications Magazine*, vol. 39, n. 3, pp. 142–148, 2001.

[18] L. Dittmann, C. Develder, D. Chiaroni et al., "The European IST project DAVID: A viable approach toward optical packet switching", *IEEE Journal on Selected Areas in Communications*, vol. 21, n. 7, pp. 1026–1040, 2003.

[19] DJ Blumenthal, "Optical packet switching", in *Lasers and Electro-Optics Society (LEOS 2004), The 17th Annual Meeting of the IEEE*, volume 2, 2004.

[20] J.S. Turner, "Terabit burst switching", *Journal of High Speed Networks*, vol. 8, n. 1, pp. 3–16, 1999.

[21] C. Qiao and M. Yoo, "Optical burst switching (OBS)–a new paradigm for an Optical Internet", *Journal of High Speed Networks*, vol. 8, n. 1, pp. 69–84, 1999.

[22] C. Qiao and M. Yoo, "Choices, features and issues in optical burst switching", *Optical Networks Magazine*, vol. 1, n. 2, pp. 36–44, 2000.

[23] S. Verma, H. Chaskar and R. Ravikanth, "Optical burst switching: a viable solution for terabit IP backbone", *IEEE Network*, vol. 14, n. 6, pp. 48–53, 2000.

[24] Y. Chen, C. Qiao and X. Yu, "Optical burst switching: A new area in optical networking research", *IEEE Network*, vol. 18, n. 3, pp. 16–23, 2004.

[25] A.M. Odlyzko, "Internet traffic growth: Sources and implications", in *Proc. SPIE*, volume 5247, pp. 1–15, 2003.

[26] J. Davidson, J. Peters, M. Bhatia, S. Kalidindi and S. Mukherjee, *Voice over IP Fundamentals (Fundamentals)*, Cisco Press, 2006.

[27] DE Smith, "IP TV bandwidth demand: Multicast and channel surfing", in *IEEE INFOCOM 2007. 26th IEEE International Conference on Computer Communications*, pp. 2546–2550, 2007.

[28] F. Thouin and M. Coates, "Video-on-demand networks: design approaches and future challenges", *IEEE Network*, vol. 21, n. 2, pp. 42–48, 2007.

[29] R. Schollmeier et al., "A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications", *Peer-to-Peer Computing*, pp. 101–102, 2001.

[30] D. H. K. Tsang, K. W. Ross, P. Rodriguez and J. Li, "Advances in peer-to-peer streaming systems", *IEEE Journal on Selected Areas in Communications*, vol. 25, n. 9, pp. 1609–1611, 2007, Guest editorial.

[31] R. Ballart and Y.C. Ching, "SONET: Now it's the standard optical network", *IEEE Communications Magazine*, vol. 27, n. 3, pp. 8–15, 1989.

[32] ITU Rec. G.707/Y.1322, "Network node interface for the synchronous digital hierarchy (SDH)", International Telecommunication Union, ITU-T, October 2003.

[33] W.N. Waggener and B. Waggener, *Pulse code modulation techniques: with applications in communications and data recording*, Chapman & Hall, 1995.

[34] J. Manchester, J. Anderson, B. Doshi and S. Dravida, "IP over Sonet", *IEEE Communications Magazine*, vol. 36, n. 5, pp. 136–142, 1998.

[35] J.T. Buckwalter, *Frame relay: technology and practice*, Addison-Wesley Professional, 1999.

[36] C. Brown and A. Malis, "Multiprotocol Interconnect over Frame Relay", Internet Engineering Task Force (IETF), RFC 2427, November 1998.

[37] X.25, "Interface between data terminal Equipment (DTE) and Data Circuit Terminating Equipment (DCE) for terminals operating in the packet mode on public data networks", *CCITT Recommendation*, vol. VIII, 1996.

[38] D. E. McDysan and D. L. Spohn, *ATM theory and applications*, McGraw-Hill Professional, 1998.

[39] R.O. Onvural, *Asynchronous transfer mode networks: performance issues*, Artech House, Inc., 1995.

[40] M. Laubach and J. Halpern, "Classical ip and arp over atm", Internet Engineering Task Force (IETF), RFC 2225, April 1998.

[41] R. Cole, D. Shur and C. Villamizar, "IP over ATM: A Framework Document", Internet Engineering Task Force (IETF), RFC 2427 (Informational), 1996.

[42] B.L. Tierney, W.E. Johnston, J.R. Lee and G. Hoo, "Performance Analysis in High-Speed Wide Area IP-over-ATM Networks: Top-to-Bottom End-to-End Monitoring", *IEEE Network*, vol. 10, n. 3, pp. 26–39, 1996.

[43] M. Maher, "ATM Signalling Support for IP over ATM - UNI Signalling 4.0 Update", Internet Engineering Task Force (IETF), RFC 2331 (Proposed Standard), 1998.

[44] E. Rosen, A. Viswanathan and R. Callon, "Multiprotocol label switching architecture", Internet Engineering Task Force (IETF), RFC 3031, 2001.

[45] B. Davie and Y. Rekhter, *MPLS: technology and applications*, Morgan Kaufmann, 2000.

[46] N. Ghani, S. Dixit and T.S. Wang, "On IP-over-WDM integration", *IEEE Communications Magazine*, vol. 38, n. 3, pp. 72–84, 2000.

[47] A. Jourdan, D. Chiaroni, E. Dotaro, G.J. Eilenberger, F. Masetti and M. Renaud, "The perspective of optical packet switching in IP dominant backbone and metropolitan networks", *IEEE Communications Magazine*, vol. 39, n. 3, pp. 136–141, 2001.

[48] N. Ghani, S. Dixit and T. Wang, "On IP-over-WDM Integration: A Retrospective", *IEEE Communications Magazine*, vol. 41, n. 9, pp. 42–45, 2003.

[49] S. Dixit, *IP over WDM: building the next-generation optical internet*, Wiley-Blackwell, 2003.

[50] Cisco, "Converge IP and DWDM Layers in the Core Network", *Cisco White paper*, 2007.

[51] A. Vukovic, "communication network power efficiency-assessment, limitations and directions", *Electronics Cooling*, vol. 10, n. 3, pp. 18, 2004.

[52] R. Dutta and G.N. Rouskas, "A survey of virtual topology design algorithms for wavelength routed optical networks", *Optical Networks Magazine*, vol. 1, n. 1, pp. 73–89, 2000.

[53] 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 (TON)*, vol. 9, n. 2, pp. 186–198, 2001.

[54] AS Arora, S. Subramaniam and H.A. Choi, "Logical topology design for linear and ring optical networks", *IEEE Journal on Selected Areas in Communications*, vol. 20, n. 1, pp. 62–74, 2002.

[55] H. Zang, J.P. Jue and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks", *Optical Networks Magazine*, vol. 1, n. 1, pp. 47–60, 2000.

[56] JQ Hu and B. Leida, "Traffic grooming, routing, and wavelength assignment in optical WDM mesh networks", in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, 2004.

[57] L.W. Chen and E. Modiano, "Efficient routing and wavelength assignment for reconfigurable WDM ring networks with wavelength converters", *IEEE/ACM Transactions on Networking (TON)*, vol. 13, n. 1, pp. 186, 2005.

[58] T. Hindam, "Solving the routing and wavelength assignment problem in WDM networks for future planning", *IEEE Communications Magazine*, vol. 47, n. 8, pp. 35–41, 2009.

[59] I. Chlamtac, A. Ganz and G. Karmi, "Lightnets: Topologies for high-speed optical networks", *Journal of Lightwave Technology*, vol. 11, n. 5, pp. 951–961, 1993.

[60] R. Ramaswami and K.N. Sivarajan, "Design of logical topologies for wavelength-routed all-optical networks", in *IEEE INFOCOM*, pp. 1316–1316, 1995.

[61] J.M.H. Elmirghani and H.T. Mouftah, "All-optical wavelength conversion: Technologies and applications in DWDM networks", *IEEE Communications Magazine*, vol. 38, n. 3, pp. 86–92, 2000.

[62] X. Chu, B. Li and I. Chlamtac, "Wavelength converter placement under different RWA algorithms in wavelength-routed all-optical networks", *IEEE Transactions on Communications*, vol. 51, n. 4, pp. 607–617, 2003.

[63] S. Subramaniam, M. Azizoğlu and A.K. Somani, "On optiml converter placement in wavelength-routed networks", *IEEE/ACM Transactions on Networking (TON)*, vol. 7, n. 5, pp. 754–766, 1999.

[64] K. Zhu, H. Zhu and B. Mukherjee, *Traffic grooming in optical WDM mesh networks*, Springer-Verlag New York Inc., 2005.

[65] S. Huang, R. Dutta and GN Rouskas, "Traffic grooming in path, star, and tree networks: Complexity, bounds, and algorithms", *IEEE Journal on Selected Areas in Communications*, vol. 24, n. 4, pp. 82, 2006.

[66] R. Dutta, A.E. Kamal and G.N. Rouskas, *Traffic Grooming for Optical Networks: Foundations and Techniques*, Springer-Verlag, 2008.

[67] B. Chen, GN Rouskas and R. Dutta, "On hierarchical traffic grooming in WDM networks", *IEEE/ACM Transactions on Networking (TON)*, vol. 16, n. 5, pp. 1226–1238, 2008.

[68] X. Yang and B. Ramamurthy, "Dynamic routing in translucent WDM optical networks: the intradomain case", *Journal of Lightwave Technology*, vol. 23, n. 3, pp. 955, 2005.

[69] N. Sambo, N. Andriolli, A. Giorgetti, L. Valcarenghi, I. Cerutti, P. Castoldi and F. Cugini, "GMPLS-controlled dynamic translucent optical networks", *IEEE Network*, page 35, 2009.

[70] L. Xu, H.G. Perros, G. Rouskas et al., "Techniques for optical packet switching and optical burst switching", *IEEE Communications Magazine*, vol. 39, n. 1, pp. 136–142, 2001.

[71] TS El-Bawab and J.D. Shin, "Optical packet switching in core networks: between vision andreality", *IEEE Communications Magazine*, vol. 40, n. 9, pp. 60–65, 2002.

[72] G.N. Rouskas and L. Xu, *Optical WDM Networks: Past Lessons and Path Ahead, K. Sivalingam and S. Subramaniam, eds.*, ch. Optical packet switching, Springer-Verlag, 2004.

[73] D.J. Blumenthal, B.E. Olsson, G. Rossi, T.E. Dimmick, L. Rau, M. Anovi, O. Lavrova, R. Doshi, O. Jerphagnon, J.E. Bowers et al., "All-optical label swapping networks and technologies", *Journal of Lightwave Technology*, vol. 18, n. 12, pp. 2058, 2000.

[74] E. Kehayas et al., "All-optical network subsystems using integrated SOA-based optical gates and flip-flops for label-swapped networks", *IEEE Photonics Technology Letters*, vol. 18, n. 16, pp. 1750–1752, 2006.

[75] Q. Xu and M. Lipson, "All-optical logic based on silicon micro-ring resonators", *Optics Express*, vol. 15, n. 3, pp. 924–929, 2007.

[76] J. Berthold, A.A.M. Saleh, L. Blair and J.M. Simmons, "Optical networking: past, present, and future", *Journal of Lightwave Technology*, vol. 26, n. 9, pp. 1104–1118, 2008.

[77] S.R. Amstutz, "Burst switching-an introduction", *IEEE Communications Magazine*, vol. 21, n. 8, pp. 36–42, 1983.

[78] S.R. Amstutz, "Burst switching-an update", *IEEE Communications Magazine*, vol. 27, n. 9, pp. 50–57, 1989.

[79] I. de Miguel, J.C. Gonzalez, T. Koonen, R. Duran, P. Fernandez and I.T. Monroy, "Polymorphic architectures for optical networks and their seamless evolution towards next generation networks", *Photonic Network Communications*, vol. 8, n. 2, pp. 177–189, 2004.

[80] C.M. Gauger, PJ Kuhn, EV Breusegem, M. Pickavet and P. Demeester, "Hybrid optical network architectures: bringing packets and circuits together", *IEEE Communications Magazine*, vol. 44, n. 8, pp. 36–42, 2006.

[81] Rishi Sinha, Christos Papadopoulos and John Heidemann, "Internet Packet Size Distributions: Some Observations", Technical Report ISI-TR-2007-643, USC/Information Sciences Institute, May 2007.

[82] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely and C. Diot, "Packet-level traffic measurements from the Sprint IP backbone", *IEEE Network*, vol. 17, n. 6, pp. 6–16, 2003.

[83] H. Buchta, *Analysis of physical constraints in an optical burst switching network*, PhD thesis, Technische Universitaet Berlin, Germany, 2005.

[84] Yijun Xiong, Marc Vandenhoute and Hakki C. Cankaya, "Control architecture in optical burst-switched WDM networks", *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1838–1851, 2000.

[85] X. Cao, J. Li, Y. Chen and C. Qiao, "Assembling TCP/IP packets in optical burst switched networks", in *IEEE Globecom*, volume 3, pp. 2808–2812, 2002.

[86] V. M. Vokkarane, Q. Zhang, J. P. Jue and B. Chen, "Generalized burst assembly and scheduling techniques for QoS support in optical burst-switched networks", in *IEEE Globecom*, volume 3, pp. 2747–2751, 2002.

[87] K. Long, R.S. Tucker and C. Wang, "A new framework and burst assembly for IP DiffServ over optical burst switching networks", in *Proc. IEEE Globecom*, pp. 3159–3164, 2003.

[88] X. Yu, J. Li, X. Cao, Y. Chen and C. Qiao, "Traffic statistics and performance evaluation in optical burst switched networks", *Journal of lightwave technology*, vol. 22, n. 12, pp. 2722, 2004.

[89] K. Christodoulopoulos, E. Varvarigos and K. Vlachos, "A new burst assembly scheme based on the average packet delay and its performance for TCP traffic", *Optical Switching and Networking*, vol. 4, n. 3-4, pp. 200–212, 2007.

[90] H. Jeong, J.Y. Choi and M. Kang, "An adaptive load-aware burst assembly scheme to achieve optimal performance of FDL buffers in OBS networks", *Photonic Network Communications*, vol. 17, n. 3, pp. 238–244, 2009.

[91] S. Lam, "Store-and-forward buffer requirements in a packet switching network", *IEEE Transactions on Communications [legacy, pre-1988]*, vol. 24, n. 4, pp. 394–403, 1976.

[92] M. Yoo and C. Qiao, "New optical burst-switching protocol for supporting quality of service", in *Proceedings of SPIE*, volume 3531, page 396, 1998.

[93] M. Yoo and C. Qiao, "Supporting multiple classes of services in IP over WDM networks", in *IEEE Globecom*, pp. 1023–1027, 1999.

[94] M. Yoo, C. Qiao and S. Dixit, "QoS performance of optical burst switching in IP-over-WDM networks", *IEEE Journal on Selected Areas in Communications*, vol. 18, n. 10, pp. 2062–2071, 2000.

[95] K. Dolzer and C. Gauger, "On burst assembly in optical burst switching networks-a performance evaluation of Just-Enough-Time", in *Proceedings of the 17th International Teletraffic Congress (ITC 17)*, 2001.

[96] M. Yoo and C. Qiao, "Just-enough-time (JET): a high speed protocol for bursty trac in optical networks,"", *IEEE/LEOS Technologies for a Global information Infrastructure*, pp. 26–27, 1997.

[97] J.Y. Wei and R.I. McFarland, "Just-in-time signaling for WDM optical burst switching networks", *Journal of Lightwave Technology*, vol. 18, n. 12, pp. 2019, 2000.

[98] K. Dolzer, "Assured horizon-a new combined framework for burst assembly and reservation in optical burst switched networks", in *Proceedings of the European Conference on Networks and Optical Communications (NOC)*, 2002.

[99] M. Dueser and P. Bayvel, "Analysis of a dynamically wavelength-routed optical burst switched network architecture", *Journal of Lightwave Technology*, vol. 20, n. 4, 2002.

[100] A.H. Zaim, I. Baldine, M. Cassada, G.N. Rouskas, H.G. Perros and D. Stevenson, "JumpStart just-in-time signaling protocol: a formal description using extended finite state machines", *Optical Engineering*, vol. 42, pp. 568, 2003.

[101] I. Baldine, H. Perros, G. Rouskas and D. Stevenson, "JumpStart: A just-in-time signaling architecture for WDM burst-switched networks", *NETWORKING 2002*, pp. 1081–1086, 2006.

[102] J. Li, C. Qiao, J. Xu and D. Xu, "Maximizing throughput for optical burst switching networks", *IEEE/ACM Transactions on Networking (TON)*, vol. 15, n. 5, pp. 1176, 2007.

[103] J. Teng and G.N. Rouskas, "A Detailed Analysis and Performance Comparison of Wavelength Reservation Schemes for Optical Burst Switched Networks?", *Photonic Network Communications*, vol. 9, n. 3, pp. 311–335, 2005.

[104] S. Yao, B. Mukherjee, S.J.B. Yoo and S. Dixit, "A unified study of contention-resolution schemes in optical packet-switched networks", *Journal of Lightwave Technology*, vol. 21, n. 3, pp. 672, 2003.

[105] H. Akimaru and K. Kawashima, *Teletraffic: theory and applications*, Springer-Verlag, 1993.

[106] S. J. B. Yoo, "Wavelength conversion technologies for WDM network applications", *Journal of Lightwave Technology*, vol. 14, n. 6, pp. 955–966, 1996.

[107] T. Durhuus, B. Mikkelsen, C. Joergensen, S. Lykke Danielsen and KE Stubkjaer, "All-optical wavelength conversion by semiconductor optical amplifiers", *Journal of Lightwave Technology*, vol. 14, n. 6, pp. 942–954, 1996.

[108] B. Ramamurthy and B. Mukherjee, "Wavelength conversion in WDM networking", *IEEE Journal on Selected Areas in Communications*, vol. 16, n. 7, pp. 1061–1073, 1998.

[109] Y. Mingwu, L. Zengji and W. Aijun, "Accurate and approximate evaluations of asynchronous tunable-wavelength-converter sharing schemes in optical burst-switched networks", *Journal of Lightwave Technology*, vol. 23, n. 10, pp. 2807, 2005.

[110] X. Chu, J. Liu and Z. Zhang, "Analysis of sparse-partial wavelength conversion in wavelength-routed WDM networks", in *IEEE INFOCOM*, volume 2, pp. 1363–1371, 2004.

[111] N. Akar, E. Karasan and K. Dogan, "Wavelength converter sharing in asynchronous optical packet/burst switching: An exact blocking analysis for Markovian arrivals", *IEEE Journal on Selected Areas in Communications*, vol. 24, n. 12, pp. 69–80, 2006.

[112] V. Eramo, M. Listanti and P. Pacifici, "A comparison study on the number of wavelength converters needed in synchronous and asynchronous all-optical switching architectures", *Journal of Lightwave Technology*, vol. 21, n. 2, pp. 340, 2003.

[113] C.M. Gauger, "Performance of converter pools for contention resolution in optical burst switching", in *Proceedings of SPIE*, volume 4874, page 109, 2002.

[114] D.K. Hunter, M.C. Chia and I. Andonovic, "Buffering in optical packet switches", *Journal of Lightwave Technology*, vol. 16, n. 12, pp. 2081, 1998.

[115] F. Masetti, P. Gavignet-Morin, D. Chiaroni and G. Da Loura, "Fiber delay lines optical buffer for ATM photonic switching applications", in *IEEE INFOCOM*, pp. 935–935, 1993.

[116] I. Chlamtac, A. Fumagalli, LG Kazovsky, P. Melman, W.H. Nelson, P. Poggiolini, M. Cerisola, A. Choudhury, T.K. Fong, R.T. Hofmeister et al., "CORD: Contention resolution by delay lines", *IEEE Journal on Selected Areas in Communications*, vol. 14, n. 5, pp. 1014–1029, 1996.

[117] D. K. Hunter, W. D. Cornwell, T. H. Gilfedder, A. Franzen and I. Andonovic, "SLOB: A switch with large optical buffers for packet switching", *Journal of Lightwave Technology*, vol. 16, n. 10, pp. 1725, 1998.

[118] F. Callegati, "Optical buffers for variable length packets", *IEEE Communications Letters*, vol. 4, n. 9, pp. 292–294, 2000.

[119] J. Aracil and F. Callegati (editors), *Enabling Optical Internet with Advanced Network Technologies*, Springer-Verlag, 2009.

[120] L. Tancevski, S. Yegnanarayanan, G. Castanon, L. Tamil, F. Masetti and T. McDermott, "Optical routing of asynchronous, variable length packets", *IEEE Journal on Selected Areas in Communications*, vol. 18, n. 10, pp. 2084–2093, 2000.

[121] M. C. Chia, D. K. Hunter, I. Andonovic, P. Ball, I. Wright, S. P. Ferguson, K. M. Guild and M. J. O'Mahony, "Packet loss and delay performance of feedback and feed-forward arrayed-waveguide gratings-based optical packet switches with WDM inputs-outputs", *Journal of Lightwave Technology*, vol. 19, pp. 19–9, 2001.

[122] I. Chlamtac, A. Fumagalli and C.J. Suh, "Multibuffer delay line architectures for efficient contention resolution in optical switching nodes", *IEEE Transactions on Communications*, vol. 48, n. 12, pp. 2089–2098, 2000.

[123] S. Bregni, A. Pattavina and G. Vegetti, "Architectures and performance of AWG-based optical switching nodes for IP networks", *IEEE Journal on Selected Areas in Communications*, vol. 21, n. 7, pp. 1113–1121, 2003.

[124] F. Callegati, W. Cerroni, C. Raffaelli and P. Zaffoni, "Wavelength and time domain exploitation for QoS management in optical packet switches", *Computer Networks*, vol. 44, n. 4, pp. 569–582, 2004.

[125] S. Yao, B. Mukherjee and S. Dixit, "Plato: A generic modeling technique for optical packet-switched networks", *International Journal of Wireless and Optical Communications*, vol. 1, pp. 91–101, 2003.

[126] F. Callegati, W. Cerroni, G. Corazza, C. Develder, M. Pickavet and P. Demeester, "Scheduling algorithms for a slotted packet switch with either fixed or variable length packets", *Photonic Network Communications*, vol. 8, n. 2, pp. 163–176, 2004.

[127] T. Zhang, K. Lu and J. P. Jue, "Shared fiber delay line buffers in asynchronous optical packet switches", *IEEE Journal on Selected Areas in Communications*, vol. 24, n. 4, pp. 118, 2006.

[128] A.G. Fayoumi and A.P. Jayasumana, "A surjective-mapping based model for optical shared-buffer cross-connect", *IEEE/ACM Transactions on Networking (TON)*, vol. 15, n. 1, pp. 226–233, 2007.

[129] J. Yang, "Optical Communications Packet scheduling algorithms for optical packet switch with shared WDM buffers", *European Transactions on Telecommunications*, vol. 20, n. 5, pp. 508–512, 2009.

[130] L. Tancčevski, A. Ge, G. Castanon and L. Tamil, "A new scheduling algorithm for asynchronous variable length IP traffic incorporating void filling", *Proc. OFC/IOOC Feb*, pp. 21–26, 1999.

[131] S. Gunreben and G. Hu, "A multi-layer analysis of reordering in optical burst switched networks", *IEEE Communications Letters*, vol. 11, n. 12, pp. 1013–1015, 2007.

[132] X. Wang, H. Morikawa and T. Aoyama, "Burst optical deflection routing protocol for wavelength routing WDM networks", *Optical Networks Magazine*, vol. 3, n. 6, pp. 12–19, 2002.

[133] C. F. Hsu, T. L. Liu and N. F. Huang, "Performance analysis of deflection routing in optical burst-switched networks", in *IEEE INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, volume 1, 2002.

[134] Y. Chen, H. Wu, D. Xu and C. Qiao, "Performance analysis of optical burst switched node with deflection routing", in *IEEE International Conference on Communications, 2003. ICC'03*, pp. 1355–1359, 2003.

[135] A.A. Amin, M. Takenaka, T. Tanemura, K. Shimizu, R. Inohara, K. Nishimura, M. Usami, Y. Takita, Y. Kai, H. Onaka et al., "Experimental Validation of Deflection Routing in a 3-Node Optical Burst Core Network with 40Gb/s Edge Nodes", *ECOC 07, Berlin, Germany*, October 2007.

[136] A. Zalesky, H. Le Vu, Z. Rosberg, E.W.M. Wong and M. Zukerman, "OBS Contention Resolution Performance", *Performance Evaluation*, vol. 64, n. 4, pp. 357–373, 2007.

[137] O. Pedrola, S. Rumley, D. Careglio, M. Klinkowski, P. Pedroso, J. Solé-Pareta and C. Gaumier, "A performance survey on deflection routing techniques for OBS networks", in *Proceedings of IEEE International Conference on Transparent Optical Networks, ICTON2009, Island of São Miguel, Azores, Portugal*, 2009.

[138] A. Zalesky, H.L. Vu, Z. Rosberg, E.W.M. Wong and M. Zukerman, "Stabilizing deflection routing in optical burst switched networks", *IEEE Journal on Selected Areas in Communications*, vol. 25, n. 6, pp. 3–19, 2007.

[139] C.M. Gauger, M. Kohn and J. Scharf, "Comparison of contention resolution strategies in OBS network scenarios", in *Proceedings of 2004 6th International Conference on Transparent Optical Networks, 2004*, pp. 18–21, 2004.

[140] C.M. Gauger, "Optimized combination of converter pools and FDL buffers for contention resolution in optical burst switching", *Photonic Network Communications*, vol. 8, n. 2, pp. 139–148, 2004.

[141] V. M. Vokkarane and J. P. Jue, "Prioritized burst segmentation and composite burst-assembly techniques for QoS support in optical burst-switched networks", *IEEE Journal on Selected Areas in Communications*, vol. 21, n. 7, pp. 1198–1209, 2003.

[142] V. M. Vokkarane and J. P. Jue, "Segmentation-based nonpreemptive channel scheduling algorithms for optical burst-switched networks", *Journal of Lightwave Technology*, vol. 23, n. 10, pp. 3125, 2005.

[143] F. Farahmand, Q. Zhang and J. P. Jue, "A closed-loop rate-based contention control for optical burst switched networks", in *IEEE Global Telecommunications Conference, GLOBECOM'05*, volume 4, 2005.

[144] V. M. Vokkarane and J. P. Jue, "Dynamic congestion-based load balanced routing in optical burst-switched networks", in *IEEE Global Telecommunications Conference, GLOBECOM'03*, 2003.

[145] L. Yang and G. N. Rouskas, "Path switching in optical burst switched networks", *Proceedings of Networking 2005*, pp. 406–418, 2005.

[146] L. Yang and G. N. Rouskas, "Adaptive path selection in OBS networks", *Journal of Lightwave Technology*, vol. 24, n. 8, pp. 3002–3011, 2006.

[147] Y. Liu, G. Mohan, K. C. Chua and J. Lu, "Multipath traffic engineering in WDM optical burst switching networks", *IEEE Transactions on Communications*, vol. 57, n. 4, pp. 1099–1108, 2009.

[148] K. Laevens, "Traffic characteristics inside optical burst-switched networks", in *Proceedings of SPIE Opticomm*, volume 4874, page 137, 2002.

[149] X. Yu, Y. Chen and C. Qiao, "A study of traffic statistics of assembled burst traffic in optical burst switched networks", in *Proc. Opticomm*, volume 4874, pp. 149–159, 2002.

[150] M. de Vega Rodrigo and J. Götz, "An analytical study of optical burst switching aggregation strategies", in *Proceedings of the Third International Workshop on Optical Burst Switching (WOBS)*, 2004.

[151] X. Mountrouidou and H. G. Perros, "Characterization of the burst aggregation process in optical burst switching", *Lecture Notes in Computer Science*, vol. 3976, pp. 752, 2006.

[152] T. Karagiannis, M. Molle, M. Faloutsos and A. Broido, "A nonstationary Poisson view of Internet traffic", in *IEEE INFOCOM*, volume 3, pp. 1558–1569, 2004.

[153] AM Law and WD Kelton, *Simulation modeling and analysis*, McGraw-Hill, 2000.

[154] A. Varga, "OMNeT++ Discrete event simulation system. User Manual", *Technical University of Budapest, Dept. of Telecommunications*, 2006.

[155] M. Izal and J. Aracil, "On the influence of self-similarity on optical burst switching traffic", in *IEEE GLOBECOM*, volume 3, pp. 2308–2312, 2002.

[156] A. Ge, F. Callegati and L.S. Tamil, "On optical burst switching and self-similar traffic", *IEEE Communications Letters*, vol. 4, n. 3, pp. 98–100, 2000.

[157] M. Izal, J. Aracil, D. Morato and E. Magana, "Delay-throughput curves for timer-based obs burstifiers with light load", *Journal of Lightwave Technology*, vol. 24, n. 1, pp. 277–285, 2006.

[158] G. B. Figueiredo, N. L. S. da Fonseca, C. A. V. Melo and M. R. Salvador, "On the transformation of multifractal traffic at ingress optical burst switches", in *IEEE ICC*, 2006.

[159] X. Mountrouidou and H. Perros, "On traffic modeling of burst aggregation algorithms using video and data traces", in *GridNets*, 2007.

[160] N. M. Garcia, P. P. Monteiro and M. M. Freire, "Burst Assembly with Real IPv4 Data-Performance Assessement of Three Assembly Algorithms", *Lecture Notes in Computer Science*, vol. 4003, pp. 223–234, 2006.

[161] P. Abry and D. Veitch, "Wavelet analysis of long-range-dependent traffic", *IEEE Transactions on Information Theory*, vol. 44, n. 1, pp. 2–15, 1998.

[162] Perl, "The Perl Programming Language", *https://www.perl.org*.

[163] N. R. Draper and H. Smith, *Applied regression analysis*, Probability and Statistics, John Wiley & Sons, 1998.

[164] H. R. van As, "Media access techniques: The evolution towards terabit/s LANs and MANs", *Computer Networks and ISDN Systems*, vol. 26, n. 6-8, pp. 603–656, 1994.

[165] N. Bouabdallah, A.L. Beylot, E. Dotaro and G. Pujolle, "Resolving the fairness issues in bus-based optical access networks", *IEEE Journal on Selected Areas in Communications*, vol. 23, n. 8, 2005.

[166] M. Pióro and D. Medhi, *Routing, flow, and capacity design in communication and computer networks*, Morgan Kaufmann Publishers, 2004.

[167] D. Eppstein, "Finding the k shortest paths", *SIAM Journal on Computing*, vol. 28, n. 2, pp. 652–673, 1999.

[168] A. Gumaste and I. Chlamtac, "Light-trails: an optical solution for IP transport", *Journal of Optical Networking*, vol. 3, n. 5, pp. 261–281, 2004, invited.

[169] J. Li and C. Qiao, "Schedule burst proactively for optical burst switched networks", in *Proc. IEEE Globecom*, 2003.

[170] G. Hu, C.M. Gauger and S. Junghans, "Performance of MAC layer and fairness protocol for the Dual Bus Optical Ring Network (DBORN)", *Proceedings of ONDM 2005*, pp. 467–476, 2005.

[171] I. Widjaja and I. Saniee, "Simplified layering and flexible bandwidth with TWIN", *Proceedings of the ACM SIGCOMM workshop on Future directions in network architecture*, pp. 13–20, 2004.

[172] Cplex, "9.0 User's Manual", *ILOG, SA*, 2003.