

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

An Assembly and Offset Assignment Scheme for Self-similar Traffic in Optical Burst Switching

Benon Kagezi Muwonge.



This thesis is submitted in fulfilment of the academic requirements

for the degree of

Master of Science in Electrical Engineering

in the Faculty of Engineering and The Built Environment

University of Cape Town

August 2006

As the candidate's supervisor, I have approved this dissertation for submission.

Name: Professor H Anthony Chan

Signed: _____

Date: _____

University of Cape Town

Declaration

I hereby declare that: (1) the above thesis is my own unaided work, both in conception and execution, and that apart from the normal guidance of my supervisor, I have received no assistance apart from that stated below; (2) except as stated below, neither the substance or any part of the thesis has been submitted in the past, or is being, or is to be submitted for a degree in the University or any other University.

I am now presenting the thesis for examination for the Degree of MSc in Electrical Engineering. I also grant the University free license to reproduce the above thesis in whole, for the purpose of research.

Muwonge Kagezi Benon

Name

Date

To my loving mother, Mrs Mary Jessy Muwonge

University of Cape Town

Abstract

Optical Burst Switching (OBS) is a viable technology for the next generation core network. We propose an FEC-assembly scheme that efficiently assembles self-similar traffic and a Pareto-offset assignment rather than a constant offset assignment. Two buffers, a packet buffer and a burst buffer, are implemented at the Label Edge Router (LER), buffering traffic in the electronic domain. The assembler, between the packet and burst buffers, is served by the packet queue while the assembler serves the burst queue. We outline advantages of why burst assembly cannot be implemented independent of offset assignment. The two schemes must be implemented in a complementary way if QoS is to be realized in an OBS network. We show that there is a direct relation between OBS network performance with burst assembly and offset assignment. We present simulation results of the assembly and offset assignment proposals using the ns2 network simulator. Our results show that the combination of the proposed FEC-Based assembly scheme with the proposed Pareto-offset assignment scheme give better network performance in terms of burst drop, resource contention and delay. Key to any traffic shaping is the nature traffic being shaped. This work also compares performance of both traditional exponential traffic with realistic Self-Similar traffic of Internet traffic on the proposed assembly and offset assignment schemes. In our simulations, we assume that all Label Switch Routers (LSR) have wavelength converters and are without optical buffers. We use Latest Available Unused Channel with Void Filling (LAUC-VF) scheduling scheme and use Just Enough Time (JET) reservation scheme.

Acknowledgements

I thank God for His guiding hand to this point of my life, for He is indeed faithful.

I wish to thank the following people for their support during the course of this research.

- My mother, and siblings, Michael and Rita Muwonge, who have been my inspiration.
- Professor H.A. Chan, my supervisor, whose continual guidance during the course of this work has taught me more than I could have ever imagined. His wisdom in many aspects of life has transformed my life.
- My colleagues and friends in the Broad Band Centre of Excellence at the University of Cape Town. The many consultations and discussions taught me a lot.

University of Cape Town

Table of Contents

Declaration	ii
Abstract	iv
Acknowledgements	v
Table of Contents	vi
Glossary	ix
List of Figures	xii
List of Tables	xvi
List of Tables	xvi
1 Introduction	1
1.1 Motivation	2
1.2 Optical Networks.....	3
1.2.1 Bottlenecks in Optical Networks.....	4
1.2.2 Solving the Bottlenecks.....	4
1.3 Overview of Optical Burst Switching.....	5
1.3.1 Problems solved by using OBS.....	6
1.3.2 Where OBS fits into the OSI model	7
1.4 Issues in Optical Burst Switching	9
1.5 Problem Statement	9
1.6 Research Objectives	10
1.7 Contribution to Knowledge	10
1.8 Scope and Limitations to this Research	11
1.9 Thesis outline	11
2 Optical Burst Switching (OBS) Fundamentals	13
2.1 A Basic OBS Network	13
2.1.1 A basic LER node architecture.....	14
2.1.2 LSR node architecture	16
2.2 Burst assembly	18
2.2.1 Threshold based assembly.....	19

2.2.2	<i>Mixed threshold assembly</i>	19
2.2.3	<i>Adaptive based assembly</i>	19
2.3	OBS traffic shaping	19
2.4	Contention resolution and burst scheduling	21
2.5	Resource reservation	29
2.5.1	<i>Tell And Go (TAG)</i>	30
2.5.2	<i>Tell And Wait (TAW)</i>	30
2.5.3	<i>Just In Time (JIT)</i>	30
2.5.4	<i>Just Enough Time (JET)</i>	32
2.6	Burst segmentation	33
<u>3</u>	<u>Previous Assembly Schemes</u>	<u>35</u>
3.1	Assembly of Packets to Bursts.....	35
3.2	General assembly schemes.....	35
3.3	Assembly with QoS specifications.....	36
<u>4</u>	<u>Proposed Assembly and Offset Assignment Scheme</u>	<u>40</u>
4.1	Design Considerations.....	41
4.1.1	<i>Considerations for burst assembly</i>	41
4.1.2	<i>Considerations for setting offset times</i>	46
4.1.3	<i>Offset assignment and burst assembly</i>	48
4.2	The packet queue	50
4.2.1	<i>Size of buffer and utilization</i>	51
4.3	Field Equivalence Classification of Packets into Bursts.....	54
4.4	The Assembler	58
4.5	The burst queue	62
4.5.1	<i>Delay tolerance</i>	63
4.5.2	<i>Class arrival rates</i>	63
4.5.3	<i>Required offset</i>	63
4.6	Assembly Queue Network.....	64
4.7	Summary of proposal	66
<u>5</u>	<u>Simulation Set-ups</u>	<u>68</u>
5.1	Simulation Set-up 1	70
5.2	Simulation set-up 2.....	71
5.3	Simulation Set-up 3	73

<u>6</u>	<u>Simulation Results and Analysis</u>	<u>75</u>
6.1	Results	75
6.2	Simulated assembly schemes	79
6.3	The Proposed FEC-based assembly scheme	81
6.4	Arrival distribution and contention at LSR	82
6.5	Offset Assignment	84
6.6	Burst Buffer Delays	86
<u>7</u>	<u>Future Work</u>	<u>89</u>
7.1	Core Aggregation Scheduling Algorithm	89
7.2	Core Network Congestion Control	90
7.3	Burst Header Packet Fields	92
<u>8</u>	<u>Conclusions and Recommendations</u>	<u>95</u>
8.1	Conclusions	95
8.2	Recommendations	96
	<u>Appendix A: The OBS-0.9a Simulator</u>	<u>98</u>
	<u>Appendix B: Timing Relationships for Circuit and Packet Switching</u>	<u>98</u>
	<u>Appendix C: Introduction to Self-Similarity</u>	<u>99</u>
	Understanding Self-Similarity	99
	The Pareto Distribution and Self-Similarity	101
	<u>Appendix D: Deriving the First Moment of Pareto Equation</u>	<u>103</u>
	<u>Appendix E: Queuing Theory Terminology</u>	<u>104</u>
	<u>Appendix F: Introduction to Decision Theory</u>	<u>105</u>
	<u>Appendix G: Random Offset Assignment Distributions</u>	<u>106</u>
	<u>References</u>	<u>108</u>

Glossary

Assemble/Burstification: The process of aggregating packets into longer packets called bursts

BHP: Burst Header Packets. These are small packets used in OBS networks to reserve network resources at LSRs. BHP packets are detached physically from the burst that they are reserving resources for

Burst: Aggregated IP packets to form a longer packet

Contention: When two or more bursts are scheduled to the same wavelength for part or all of scheduled period

Control Plane: Channels in a network, that are used solely for carrying signaling traffic

Data Plane: Channels in a network that are only used to transmit payload traffic

Deburstification: The opposite Assembly/Burstification. The process of disassembly of a burst into its individual packets at the egress node

DWDM/WDM: Dense Wavelength Division Multiplexing/ Wavelength Division Multiplexing. A technique by which two or more optical signals having different wavelengths are simultaneously transmitted in the same direction over one fibre

FDL: Fiber Delay Lines. Optical buffers that may be used in an optical network to “store” packets before switching at an LSR. Optical buffers can only delay light for very short time intervals.

FEC: Field Equivalence Classification. The classification of traffic into different QoS requirements

Ingress/Egress: Entry and exit points to a network. Same as LER

IP: Internet Protocol. A protocol that provides a connectionless service between networks.

LER: Label Edge Router. An ingress node to an OBS network, where traffic shaping is done. The LER is also the Egress node to the OBS network

LSR: Label Switch Router. A core node in an OBS network for switching bursts optically and processing the header packets electronically

OBS: Optical Burst Switching. An alternative emerging technology to either packet or circuit switching. OBS, if successful will be the intermediate technology between the current electronic switching, and the future all optical switching technologies

Optical Network: High capacity telecommunications networks based on optical technologies and components that provide routing, grooming and restoration at the wavelength level as well as wavelength-based services

O-E-O: Optical Electrical Optical Switching. The processing of optical signals by conversion from optical to electronic then optical before switching

Offset: Time period between the BHP and the Burst. It is used to compensate for processing of the BHP at every LSR

O-O-O: All optical switching The switching of optical signals optically. Future all optical networks are envisioned to have all optical processing

OSI: Open System Interconnection. A seven layer reference model in which communication functions are broken down into one of seven layers, each layer providing clearly defined services to adjacent layers

OXC: Optical Cross Connect. A node used in optical networks to switch high-speed optical signals in the optical domain

Pareto distribution: A power law probability distribution found in a large number of real world situations

Poisson distribution: A discrete probability distribution that expresses the probability of a number of events occurring in a fixed period of time

QoS: Quality of Service. The performance of a communications system expressed in terms of QoS.

Queue Network: A tandem connection of queues forming a network

SCU: Switch Control Unit. Part of an LSR used to interpret information from the header packets

Self-Similarity: An object is said to self-similar, if at different time scales, smaller or larger, the object maintains the same distribution

SONET/SDH: Synchronous Optical Network/ Synchronous Digital Hierarchy. An interface standard for synchronous optical fibre transmission, applicable to the physical layer of the OSI reference model

VOIP: Voice Over IP. A system of enabling voice data to be delivered using the IP, sometimes referred to as IP telephony

University of Cape Town

List of Figures

Fig. 1.1 An abstract performance of the OBS paradigm.....	6
Fig. 1.2 The current OSI model.	7
Fig. 1.3 The intermediate OSI model.	8
Fig. 1.4 The future all optical model.	8
Fig. 2.1 A basic mesh OBS topology with 3 LERs and 5 LSRs. The Red lines represent the control plane.	14
Fig. 2.2 A basic LER with the basic LER functionalities.....	15
Fig. 2.3 A general flow diagram for the functionalities at an LER node.....	16
Fig. 2.4 A basic LSR with the control plane and data plane separated.....	17
Fig. 2.5 Flow chart of a basic OBS LSR with the option of an FDL.....	18
Fig. 2.6 An idealistic burst arrival at an LSR with no resource contention, best achieved in Synchronous networks.....	21
Fig. 2.7 Resource contention of bursts with bandwidth not being fully utilized.....	22
Fig. 2.8 Contention resolution using FDLs. We assume no wavelength converters are available in this case.....	23
Fig. 2.9 Contention resolution using wavelength converters only.....	24
Fig. 2.10 A burst B requiring scheduling.....	25
Fig. 2.11 Burst B being scheduled using LAUC without void filling.	26
Fig. 2.12 LAUC-VF scheduling.	27

Fig. 2.13 Scheduling with Min-ev.	28
Fig. 2.14 Scheduling using the Best-fit method.	28
Fig. 2.15 Performance of JIT.	31
Fig. 2.16 JIT with release packets for each burst.....	31
Fig. 2.17 Performance of JET.....	32
Fig. 2.18 A TSOBS burst.....	33
Fig. 4.1 Proposed offset assignment scheme.	42
Fig. 4.2 The proposed addition input, output queues and ON/OFF modelling of the traffic.	43
Fig. 4.3 Packet FEC classification scheme flow chart at the LER packet buffer.	57
Fig. 4.4 Flow diagram of the proposed FEC classification, during burst assembly.	60
Fig. 4.5 Assembly procedure of the proposed FEC-assembly scheme.	61
Fig. 4.6 An LER scheduling scheme that allows for void filling at the LER.	64
Fig. 4.7 A Jackson network from which we derive the formulation of the queuing at the LER. ..	64
Fig. 4.8 The resulting LER from the proposals given in this work.	67
Fig. 5.1 A mesh Topology 1 with 4 LSRs, the smallest of the three topologies used for simulation.	73
Fig. 5.2 A mesh Topology 2 with 8 LSRs used for simulation.	74
Fig. 5.3 A mesh Topology 3 of 20 LSRs, the largest of the topologies used for simulation.....	74
Fig. 6.1. Increasing of simulation time increasing number of packets and therefore the number of bursts.....	77

Fig. 6.2 The increase in burst drop with increase in simulation time.....	78
Fig. 6.3 The ratio of bursts dropped to bursts transmitted against increase in the Hurst parameter	79
Fig. 6.4 Burst lengths when a timer based scheme is used for assembly without FEC.....	80
Fig. 6.5 Assembly with a mixed timer-length based assembly scheme.....	80
Fig. 6.6. Assembly of bursts with constant length assembly scheme with 650 packet sizes.....	81
Fig. 6.7. Distribution of assembled bursts using the proposed FEC-Based Assembly scheme.....	82
Fig. 6.8 Arrival distribution of bursts at an LSR 1, when a constant offset assignment scheme is used at the LER.....	83
Fig. 6.9 Arrival distribution of bursts at an LSR, when a Pareto distributed offset assignment scheme is used at the LER.	84
Fig. 6.10 Summary of the performance of the proposed assembly and offset assignment schemes using the proposed queue management method.	85
Fig. 6.11 Performance of the proposed FEC-Assembly scheme on both the Pareto and exponential traffic distributions and different offset assignment methods, with increase in the size topology.	86
Fig. 6.12 How delay of assembled bursts varies at the LER when different offset assignment schemes are used on the timer-based assembly scheme.	87
Fig. 6.13 How delay of assembled bursts varies at the LER when different offset assignment schemes are used on the proposed FEC-Assembly scheme.....	87
Fig. 7.1 Payload bursts arriving at LSRs A and B are routed to C. At C, the Bursts are aggregated depending on their time difference, δ , destination, d_1 , or d_2 , and/or the resources available at node C. The bursts may separate at D depending on their destinations.	89

Fig. 7.2 The paths indicated, show routes taken by BHP packets. Passing each LSR, a BHP packet records the status of each LSR at that time. On reaching the next node, this information is read. The LSRs then use this information to route traffic accordingly.....91

Fig. 7.3 The BHP shows fields appended at the ingress node, on entering the core network.92

Fig. 7.4 The BHP shows fields appended at the in the core network by an LSR, the node at which burst aggregation occurs.93

University of Cape Town

List of Tables

Table 1. highlights the desirable offset times for each of the above factors and also suggests that offset time should managed in a flexible way.	48
Table 2. showing different traffic types and their delay requirements.	49
Table 3. Packet types and their correspondence FEC preferences	55
Table 4. Packet type preferences giving relative preferences	56
Table 5. Simulation environment.	68
Table 6. Pareto and exponential traffic settings.....	70
Table 7. simulation scenarios simulated.....	71
Table 8. Traffic settings used.....	76
Table 9. Showing the Pareto and exponential traffic settings send different number of packets..	76
Table 10 Pareto traffic parameters.....	103

Chapter 1

1 Introduction

This chapter introduces the concept of Optical Burst Switching (OBS) as a viable alternative to current electronic switching, and as an intermediary technology to the future all-optical network. OBS has received tremendous attention from industry and research community [1] [2]. In this chapter we give a motivation for this research, state the issues in OBS and clearly outline the objective of this research. We then state our contributions to knowledge in this field and specify the scope and limitations of this work.

Communication in today's society is becoming faster and faster, and the demand for high-speed communication networks is now evident. Communities that do not have Internet connection are finding Internet communication a necessity to compete in today's global economy, users, with dial-up connections, are demanding broadband connection, and those with broadband Internet connection are demanding still more bandwidth. At almost all levels of society across the world, the Internet either is or is fast becoming a driving force to economic growth.

Internet communication today is vital to almost all aspects of life, and its role in people's lives increases day by day with the continual development of new applications and services on the Internet. A recent such example is Voice over IP (VoIP), which is a more affordable way for communicating by speech. Affordability of a communication using data networks is critical to the continual growth of Internet technology. For communication to have a tangible effect on society, it must be available as well as affordable to the majority or ideally all of the society.

Besides communication being basic need for any modern economy, the social impact of the continual increase of the rate of communication, and the amount of information per transmission are evident. Because of faster communication, transactions between end-users are much faster and direct and hence more workload. As an example, if customer A_1 orders from supplier X certain goods, the order by A_1 will be received within seconds. With the availability of the Internet in most parts of the world, other customers to X, $A_2, A_3, \dots, A_{1000000}$ send their

requests as well. Supplier X will (or may) get several thousands of orders on a given day. Supplier X therefore remains very busy in an attempt to satisfy his customers. This may be contrasted with orders done by post mail. Such orders are slow, and the time between orders demands less work from the supplier. This may to a large extent explain the fast pace in first world countries today, where communication is more efficient, compared to a slower pace in Africa, or third world countries in general. To facilitate the demand for faster communication, SONET/SDH was developed in Europe (SDH) and America (SONET) [3].

SONET/SDH has in recent years been what may be called the first generation fast communication technology. As most, if not all of the world's economies, become totally dependant on communication technology in the near future, SONET/SDH will not be able to handle Internet traffic, as is already apparent [3][4]. The failure is attributed to the fact that SONET/SDH was designed for voice traffic that is not rapidly variable traffic, which Internet traffic is today. SONET/SDH will have to either be improved or be replaced by a more efficient technology for more efficient communication. While work is underway to improve the SONET/SDH protocol [5], it seems certain that it might be replaced by an emerging technology offering a better alternative. This thesis is a study of one such emerging technology, as a viable alternative for the core network.

1.1 Motivation

This study is motivated by the lack thereof of a viable core network on the African continent. Communication innovations in the past have mostly come as a necessity, rather than a luxury. RADAR was for example first used in the Second World War to detect enemy aircraft by (arguably) the British [6], which in turn radically revolutionised long distance microwave communication technology. Another good example is the Internet, which started out as a way for the American Department of Defence (DoD) to transmit secrete information in a very short period of time over long distances. The results of the beginnings of the then ARPAnet by the DoD have undoubtedly changed the way the world communicates today.

A necessity exists in Africa today and in third world countries at large, and that is one of information technology to provide Internet communication. The exponential growths in the

middle class population and educated people on the continent have made communication, which was once a luxury in Africa, to be an absolute necessity for the future economical growth of the continent. The cost of building an optical core network to enhance communication capacity, is very high, and this has meant that most communication for long distances is done by airwaves, and ironically, communication in many cases from one country to another on the continent goes via an optical fibre, out of Africa (to Europe) and then back to Africa. Middle class population growth and lack of a high-speed communication core, have led Africa, and third world countries at large, to having the highest communication costs in comparison with the first world countries which exacerbates the existing information gap.

In an attempt to catch-up with the first world countries, African countries have invested heavily in legacy communication technology. Maintaining and upgrading legacy technologies is very costly and replacing the old technology with new technology is even more expensive. What Africa needs is a solution that is designed to its needs and one that can be developed and maintained “indigenously”. Though costly at first, this will prove to be the most viable means by which Africa will develop its information communication capacity to levels that are experienced in first world countries.

While Africa and third world countries in general have had little success in building a viable communication core network, first world countries have had a phenomenal success! In this thesis, we study an emerging technology in optical networks that has relevance to the communication challenges faced in Africa.

1.2 Optical Networks

The unprecedented growth in IP Internet traffic in first world countries has greatly increased the demand for high-speed transmission technologies. This has fuelled much of the research activity in optical networks, which offers the required speeds and capacity to cope with the continued growth in traffic. Advances made in Wavelength Division Multiplexing (WDM) [7] technology from the early 1990s to today (2006) have greatly surpassed all expectations. Recent advances by NEC© [8] demonstrated transmission of 273 channels each with 40Gb/s data speed for over 117km in a single fibre! A total of 10.9Tb/s! The implications of such high

capacities and speed on a single optical fibre are many. One of which is the tremendous reduction in the cost of bandwidth.

1.2.1 Bottlenecks in Optical Networks

Unfortunately, the full potential of optical fibre in optical network has not yet been realized due to a bottleneck that exists in core routers (switches) which process all received data through an optical-electrical-optical (O-E-O) conversion. This conversion means light propagation must be stopped, processed, and then transmitted, at all core nodes.

The ideal case is to have an all-optical network, in which core routers process all traffic in the optical domain, thereby not temporarily stopping traffic. However, an all-optical core network is several years from realization since optical processing is not yet a reality. To accommodate Internet growth and reduce cost in the near future, the bottlenecks must be resolved.

1.2.2 Solving the Bottlenecks

In an attempt to go round the bottlenecks at the core routers, Optical Burst Switching (OBS), now an emerging technology has been proposed as a viable solution [9].

To introduce how OBS works, we first briefly describe two existing switching technologies: circuit switching and packet switching. In circuit switching, a light path is set-up from the ingress node to the egress node. Traffic is then transmitted in the set path. During transmission of data in the set-up path, traffic is switched in the optical domain, that is, optical-optical switching (O-O). The light path is then disconnected only after transmission. In packet switching, no light path is set-up. Packets at the ingress node go through an E-O conversion and are transmitted to a core node, where they are buffered electronically, go through an O-E-O conversion and then transmitted to the next node.

Advantages of circuit switching

1. Core nodes are less expensive, since no O-E-O conversion is needed.
2. Transmission of traffic is transparent
3. Core switching is fast, and no delays are introduced at core nodes since no buffering is done.

Disadvantages of circuit switching

1. Circuit switching inefficiently uses the available bandwidth. By holding a light path from the ingress node to egress node, much of the bandwidth during transmission is idle.
2. Path set-up and disconnection is slow. As a result long delays are incurred by traffic at the ingress node.
3. Long periods of end-to-end delay.

Advantages of packet switching

1. Uses available bandwidth efficiently. Bandwidth is allocated on as needed basis.
2. Priorities can be implemented for each transmitted packet, therefore ensuring QoS for critical applications
3. Packets are accepted even when the network is busy, and queued at a core nodes

Disadvantages of packet switching

1. Core nodes are a bottleneck in packet switching. Due to the O-E-O conversion, data propagation momentarily stops at each of the core nodes. With buffers having a finite size, incoming data greatly exceeds the rate at which retransmission occurs resulting in congestion.
2. Core nodes are expensive due to the O-E-O conversion

An ideal switching technology would be one where the delays at the core nodes are not incurred, as is the case for circuit switching, and have maximum bandwidth utilization as in packet switching. This would be an all-optical network. Since to-date all-optical processing is still not practical, OBS a compromise switching paradigm, has been proposed. The circuit and packet switching paradigms are actually complementary. The disadvantages of packets switching are the advantages of circuit switching and *vis-à-vis*. The two paradigms used in a complementary way result in the OBS paradigm being studied in this work [10], [11]. Figures showing timing relations of packet and circuit switching are given in Appendix B.

1.3 Overview of Optical Burst Switching

In OBS, packets arriving at the ingress node are assembled into bursts by the aggregation of packets into a series of packets called a burst. The criteria of aggregation of packets into a burst depends on the adopted assembling scheme. The assembled burst is assigned a wavelength

and Burst Header Packet (BHP), which is transmitted ahead of the burst. BHPs are transmitted on designated out-of-band wavelengths. At each of the core nodes, a BHP reserves a wavelength for the burst. Depending on the type of reservation scheme, a wavelength reservation period will vary. After reserving a wavelength, the BHP is transmitted to the next LSR. If a BHP fails to reserve a wavelength, it is dropped and the burst dropped as well when it arrives at the LSR. This is for the case of no Fibre Delay Lines (FDLs) at all LSRs. After every O-E-O conversion of a BHP, in case of a successful reservation, the previous offset time reduces by the amount of processing time of the BHP. If a burst reaches the egress node, it is disassembled into its individual packets. Fig 1.1 shows the basic operation of OBS. By definition OBS is a one way protocol, therefore no signalling takes place before bursts are transmitted.

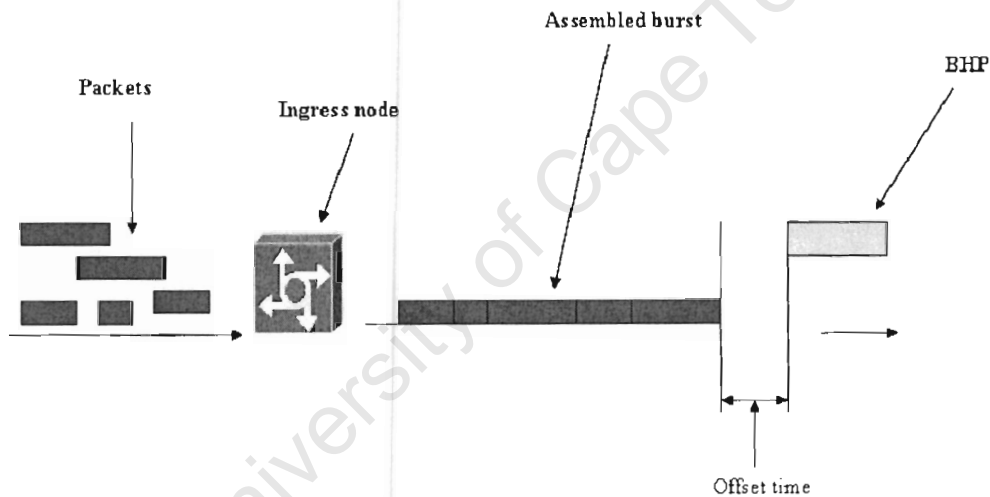


Fig. 1.1 An abstract performance of the OBS paradigm.

1.3.1 Problems solved by using OBS

OBS solves three major problems encountered today in optical and traffic engineering:

1. How to handle bursty traffic

A growing concern in recent years has been on how to handle Internet traffic which is bursty, and increasingly so, with passing years as the number of Internet users continues to grow. Current technologies are found still wanting in adequately handling bursty traffic. OBS offers a better method of solving this problem, citing the manner in which it

assembles and transmits traffic. On how to handle bursty traffic is the main focus of this study and we propose a comprehensive proposal in this regard.

2. Reduced end-to-end delays

OBS brings the communication world a step closer to the future two layered OSI model. With OBS, IP traffic can be directly transported over the physical layer using the OBS protocol. While other upper layer protocols may be used, they can be bypassed, since OBS can maintain its own QoS and handle its own routing procedures. Therefore, delay incurred by upper layers would be avoided, making transmission of information from the ingress node to the egress node much faster.

3. Cost of transmission of traffic

The use of an OBS technology would greatly reduce the costs of transmission of traffic. This reduction in cost is mainly attributed to the less overhead for the O-E-O processing needed compared to packet switching. There is more efficient use of the available DWDM wavelengths and cheap and simple LSR nodes. The cheaper means of transmission would inevitably reduce the overall cost of communication to end users. The transmission by burst assembly as opposed to circuit switching, makes available bandwidth in the fibre practically infinite. That is, it is unlikely that at a given instance, wavelengths in a fibre with over 160 wavelengths, at least 40 Gbps each, would be fully utilized. Bandwidth therefore exceeds demand, and hence much cheaper bandwidth costs.

1.3.2 Where OBS fits into the OSI model

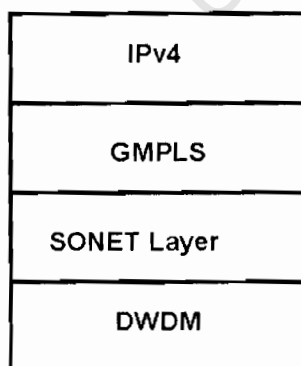


Fig. 1.2 The current OSI model.

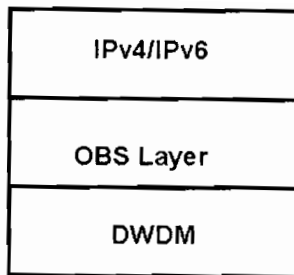


Fig. 1.3 The intermediate OSI model.

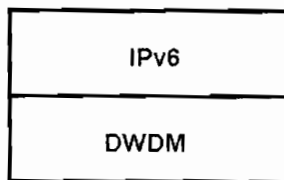


Fig. 1.4 The future all optical model.

Fig 1.2 shows the current layout of the OSI model. Fig 1.4 shows the layout of the future all-optical model. Fig 1.3 shows the model with the OBS protocol. As can be seen, the OBS model can be used with the current protocols, or without them. This makes OBS the transitional technology to the future all-optical model. On the other hand, the OBS protocol can be implemented directly below the application layer without worrying about legacy compatibility problems, which makes it ideal for implementation in third world countries. The protocol is also future proof, for the transition to an all-optical network.

However, there are challenges in the implementation of the OBS protocol. During transmission, if wavelength conversion at the LSRs is used, the assigned wavelength may change from LSR to LSR depending on the availability of resources at the time of resource reservation. Wavelength conversion capabilities of LSRs alleviate several issues in OBS such as contention resolution and at the same time introduce new issues such as increased cost of LSRs. An extreme alternative is to use buffers instead of wavelength converters. Buffers too present several challenges that are avoided by using wavelength converters complexity of LSR. In the next section we look at the challenges presented in OBS research.

1.4 Issues in Optical Burst Switching

Several issues, regarding the architecture and protocol preferences of an OBS network, are highlighted in [12] and [1] respectively, for the next generation high-speed optical network. The main concern in any design of a core network is QoS. Thus the current issues in OBS revolve around improving the QoS performance of the paradigm. Measures of QoS in OBS are delay, jitter, and burst loss. Optimizing the performance of any of these measures are still open problems. In this study, we specifically concentrate on the delay performance measure as related to burst assembly. Most of the delays that packets incur are at the LER. Optimum routing algorithms may solve jitter. Burst loss is mostly due to resource contention in the core and can be solved with improved scheduling schemes. Contention occurs when two or more bursts on an LSR with the same output port, request for the same wavelength for the duration or part of the reservation period. Each of the QoS measures cannot be completely isolated, with each one of them affecting the other to some extent.

1.5 Problem Statement

There is a need for a realistic burst assembly framework of an OBS network from which researchers in the field can base their results without having to make too many assumptions in their study. A comprehensive assembly scheme would form the foundation for the framework in OBS. The need for a common framework is critical due to the interdependence of the different schemes in an OBS network. As an example, a study on routing in an OBS network will greatly be affected by the assumptions made on the inter-arrival distribution of the bursts at the LSR, which in turn depends on assembly and transmission at the LER. As such there is a need in this field for an assembly scheme that takes into account, most if not all factors critical to burst assembly. Such a scheme would be the foundation for most research in the field, onto which scheduling and reservation schemes may be designed. Different results from different schemes would therefore be reliably compared.

There have been several assembly schemes proposed for OBS networks, and there are several that are commonly used in literature. Current assembly schemes however, do make several assumptions that make them more theoretical than practical.

This study proposes an assembly and offset assignment scheme for OBS networks. The scheme takes into account realistic factors that affect burst assembly, and thus have fewer assumptions, which makes our work more realistic. In our scheme we argue for the need to combine offset assignment to burst assembly, in contrast to past assembly schemes. In addition, our scheme takes into account many of the important factors that should be considered during assembly.

1.6 Research Objectives

This study has four main objectives:

- To create a framework for future research in OBS for the Communication and Research Group (CRG) at the University of Cape Town
- To investigate the impact of bursty traffic on OBS performance
- To propose a realistic assembly scheme with traffic queues and simulate the proposal
- To propose and simulate an offset assignment scheme that maintains QoS service of packets

1.7 Contribution to Knowledge

- We propose an assembly-offset assignment scheme for an OBS network, taking into consideration most of the critical factors of traffic that would affect QoS. The most notable difference in this scheme is the assignment of offset time at the assembly time, rather than assigning offset as a different scheme
- We present a Field Equivalence Classification Scheme (FEC) using four classes of traffic using decision theory [51]
- Another contribution of this work is the study of self-similarity and queuing as related to optical burst switching. We derive from Jackson network theory an open feed forward queuing network

1.8 Scope and Limitations to this Research

This study encompasses two other main fields of study in addition to OBS, queuing theory and self-similarity. Because the focus is on OBS, we do not go into immense detail of the mathematical analysis of queuing and self similarity. References are given for detailed discussions of the formulae and theory where appropriate. Where appropriate, we derive equations that give more understanding to a concept under consideration.

1.9 Thesis outline

This section gives the outline of the thesis.

- Chapter I

In this chapter we have presented a motivation for this research. We have outlined the basic concept of OBS, and the issues currently under research. Research objectives have been given, and contributions from this research stated. The scope and limitations of the research have been outlined as well.

- Chapter II

In this chapter we give more technical analysis of the OBS technology, defining and explaining the concepts used. We do this by categorizing OBS into three main schemes: Burst assembly, contention resolution and scheduling, and resource reservation. A later variant of OBS is described so as to make a comparison between the two protocols.

- Chapter III

In this chapter we detail review of previous assembly schemes at the LER.

- Chapter IV

In this chapter we propose an assembly and offset assignment scheme at the LER. The proposed scheme is derived from the review done in Chapter III. In the proposal we detail parameters taken into account in the design, and assumptions. The proposed scheme incorporates ideas from previous schemes with new ones to in a bid to have a realistic assembly scheme.

- Chapter V

In this chapter we detail various the simulation set-ups and settings used for the simulations. We do extensive comparisons of the performance of the proposed scheme

with both the exponential and Pareto distributed traffic. We make several settings for each traffic distribution in order to gauge the difference in performance. Settings are made for a comparison with previous assembly schemes.

- Chapter VI

In this chapter we present and discuss results from our simulations.

- Chapter VII

In this chapter we, in addition, propose a routing and scheduling scheme for LSRs to further reduce core burst drops and delay. We do not simulate this proposal.

- Chapter VIII

In this chapter we give our conclusions and recommendations from our findings in this study.

University of Cape Town

Chapter 2

2 Optical Burst Switching (OBS) Fundamentals

This chapter outlines technical aspects of OBS and issues currently being addressed.

2.1 A Basic OBS Network

The basic OBS network architecture supports payload data transparency in the entire core network, by switching in the wavelength domain. An OBS network must have out-of-band signalling, and is basically a one way protocol to minimize complexity of the core network. The payload data and the header packet must be separated by a distance called an offset time to accommodate header processing. The core network is kept as simple and as cheap as possible by putting most of the intelligence at the edge of the OBS network. Finally, the network operates in an asynchronous manner, to allow for bursty traffic to be efficiently transported into the core network.

An OBS network consists of an ingress node, called the Label Edge Router (LER), core nodes called Label Switch Routers (LSR), Dense Wavelength Division Multiplexed (DWDM) optical fibres, and an egress node, also an LER. Though the LER and LSR terminology is commonly used in MPLS/GMPLS, it has been widely adopted for OBS networks in research literature. The difference in the operation of an OBS network and a traditional optical network are the functionalities of each of the network components. Different design prototypes exist in literature, and many more proposals are still being made [15] [16]. However, all designs have several fundamental components that we outline in this section. Fig 2.1 shows a layout of the OBS network.

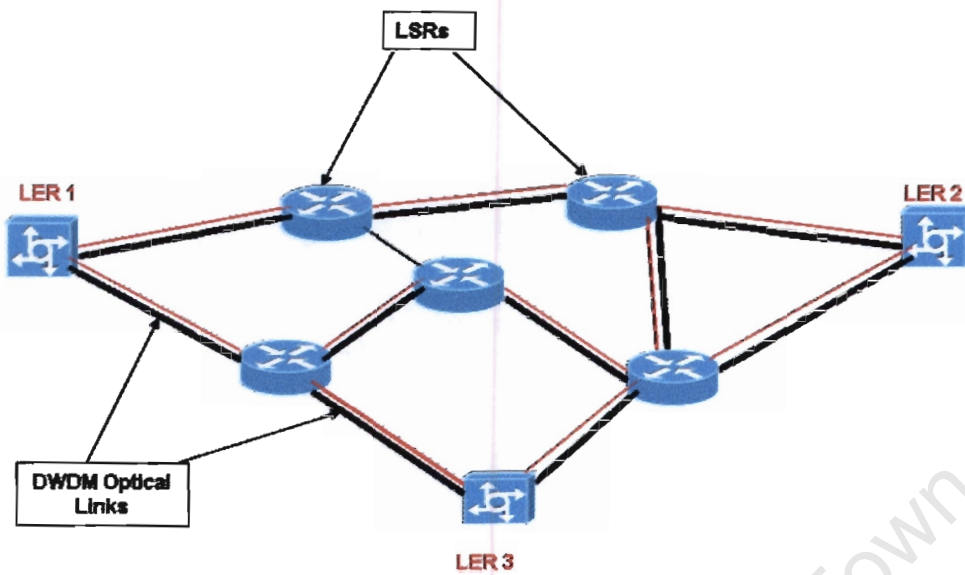


Fig. 2.1 A basic mesh OBS topology with 3 LERs and 5 LSRs. The Red lines represent the control plane.

2.1.1 A basic LER node architecture

The LERs should primarily consist of an assembler, a scheduler, classification, routing and a de-burstification module and a packet buffer. The LER receives packets in the electronic domain, and buffers them electronically. The classification module sorts the buffered packets into classes and into appropriate queues or queue according to a pre-defined FEC. The assembler then assembles the classified packets into bursts using a preferred assembly scheme. The assembled bursts are then re-buffered into a burst buffer, and scheduled for transmission using a preferred scheduling scheme. The routing module determines the route of the burst to its destination and adds the information to the BHP. At a scheduled time, the buffered burst goes through an E-O conversion and is transmitted after an offset with a corresponding BHP. Fig 2.2 shows a basic description of a basic LER.

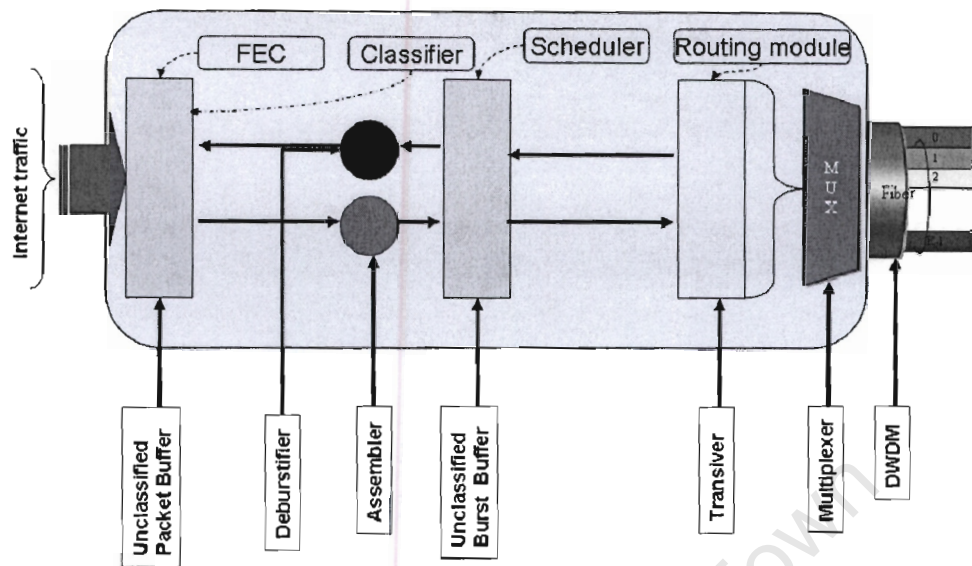


Fig. 2.2 A basic LER with the basic LER functionalities.

We will briefly define several of the functionalities at the LER:

1. Field Equivalence Classification (FEC): From the packets that arrive at the LER, while still in the electronic domain, packets are classified according to their properties. These properties include, delay tolerance, destination, packet loss tolerance and application type. Generally, when no segmentation is accommodated in the core network, all packets in a particular classification must have the same destination.
2. Assemble/Burstification: This is the process of aggregating packets that have been classified according to a specified FEC. The method of burstification may vary depending on QoS requirements. The assembled bursts are then assigned BHPs, and depending on the next available transmission time period, a burst and BHP are scheduled for transmission.
3. Buffer: In this work we always refer to an electronic buffer. At any point at the LER where packets or bursts are stationary, they are buffered.
4. Schedule: After bursts have been assembled, they must be assigned a wavelength. A wavelength may or may not be available at the instance that the burst has been burstified, or assigned an offset. Algorithms are used to schedule when a burst may be transmitted into the core network after assembly.
5. Deburstification: Deburstification is the opposite of burstification. This only happens at

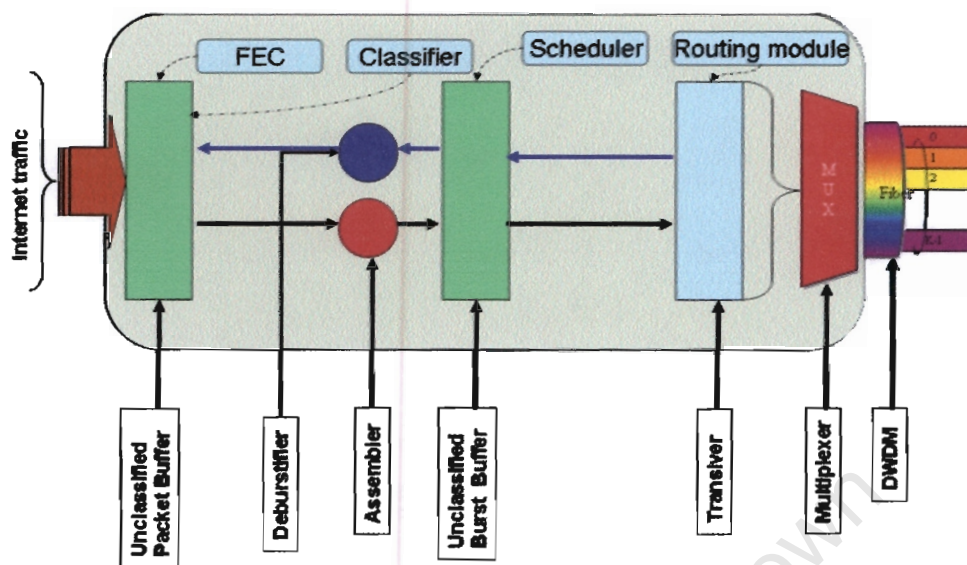


Fig. 2.2 A basic LER with the basic LER functionalities.

We will briefly define several of the functionalities at the LER:

1. Field Equivalence Classification (FEC): From the packets that arrive at the LER, while still in the electronic domain, packets are classified according to their properties. These properties include, delay tolerance, destination, packet loss tolerance and application type. Generally, when no segmentation is accommodated in the core network, all packets in a particular classification must have the same destination.
2. Assemble/Burstification: This is the process of aggregating packets that have been classified according to a specified FEC. The method of burstification may vary depending on QoS requirements. The assembled bursts are then assigned BHPs, and depending on the next available transmission time period, a burst and BHP are scheduled for transmission.
3. Buffer: In this work we always refer to an electronic buffer. At any point at the LER where packets or bursts are stationary, they are buffered.
4. Schedule: After bursts have been assembled, they must be assigned a wavelength. A wavelength may or may not be available at the instance that the burst has been burstified, or assigned an offset. Algorithms are used to schedule when a burst may be transmitted into the core network after assembly.
5. Deburstification: Deburstification is the opposite of burstification. This only happens at the

LER, and in the electronic domain. A burst reaching its destination is converted from the current optic domain into the electronic domain. The burst is then disassembled.

A general algorithm of the functionalities in an LER is highlighted in the Fig 2.3 flow chart.

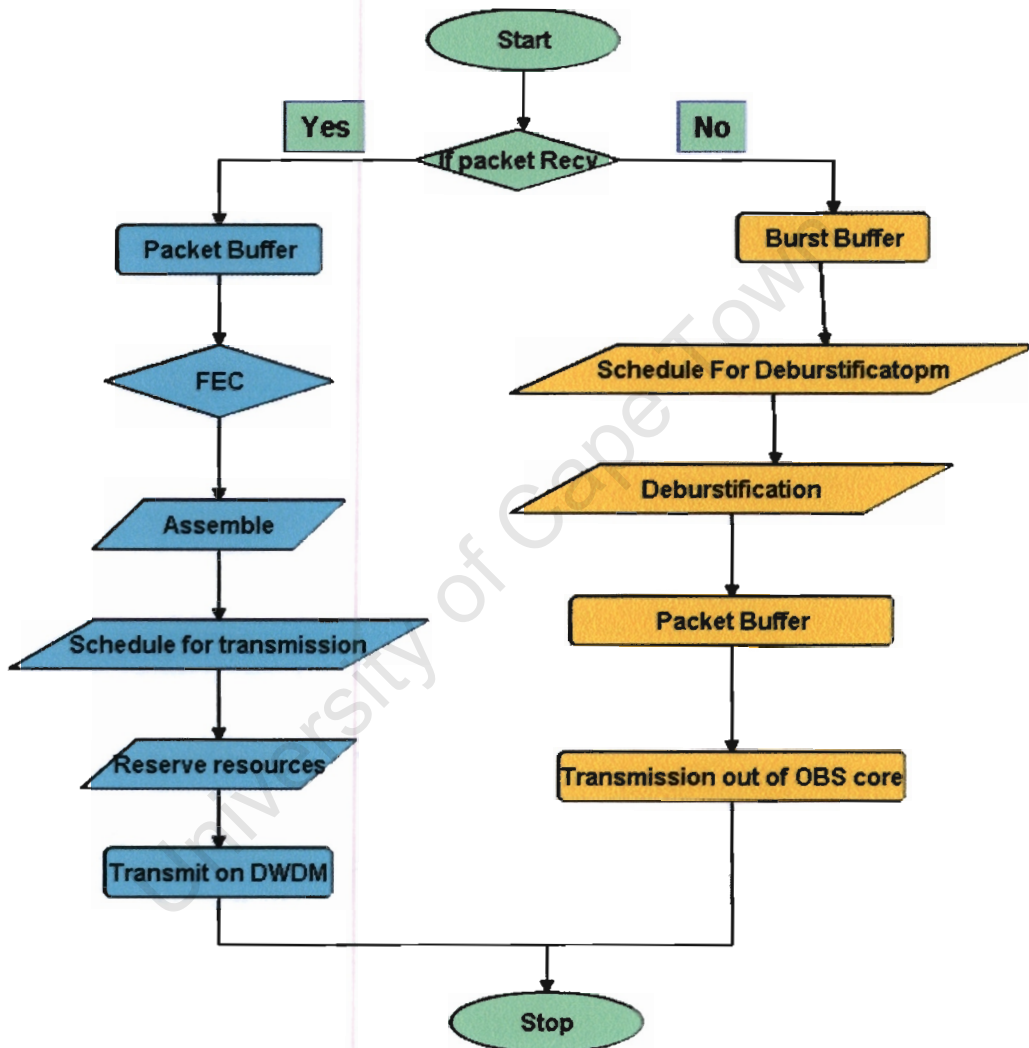


Fig. 2.3 A general flow diagram for the functionalities at an LER node.

2.1.2 LSR node architecture

The LSR primarily consists of an Optical Cross Connect (OXC) and a Switch Control Unit (SCU). The OXC operates entirely in the optical domain, while the SCU operates in the electronic domain. The SCU therefore has an OE and EO converter for the BHP. The SCU is the

intelligent part of the LSR that processes the BHP and controls the configuration of the OXC. Policies regarding burst scheduling, burst dropping are done by the SCU. In the case where the burst arrives before the BHP, the OXC will discard the burst. Should the LSR have a buffer, the SCU then is more complicated, with more processing capabilities. In this work we do not consider an LSR with a buffer for reasons we detail later. Fig 2.4 shows a basic description of an LSR.

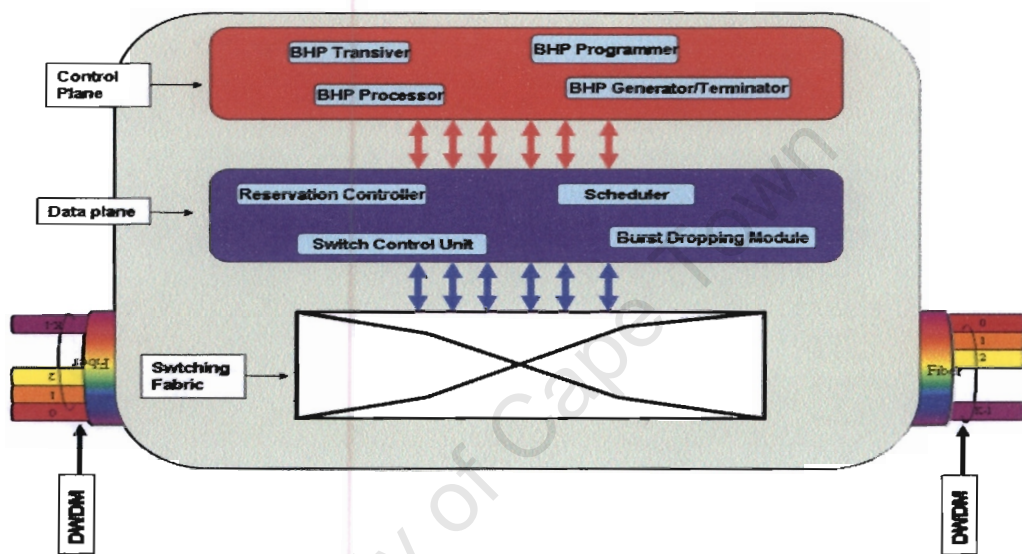


Fig. 2.4 A basic LSR with the control plane and data plane separated.

After a transmission into the core, at an LSR, a time period is scheduled for the burst from information read from the BHP. Resources are then reserved for the scheduled burst. Resources reserved for a burst may become un-available if pre-emption is enforced in case of service differentiation. Otherwise, reserved resources are held till the burst arrives at the LSR, at which point resources are allocated to a burst. Fig 2.5 shows a general flow diagram of a scheduling and reservation schemes at the LSR.

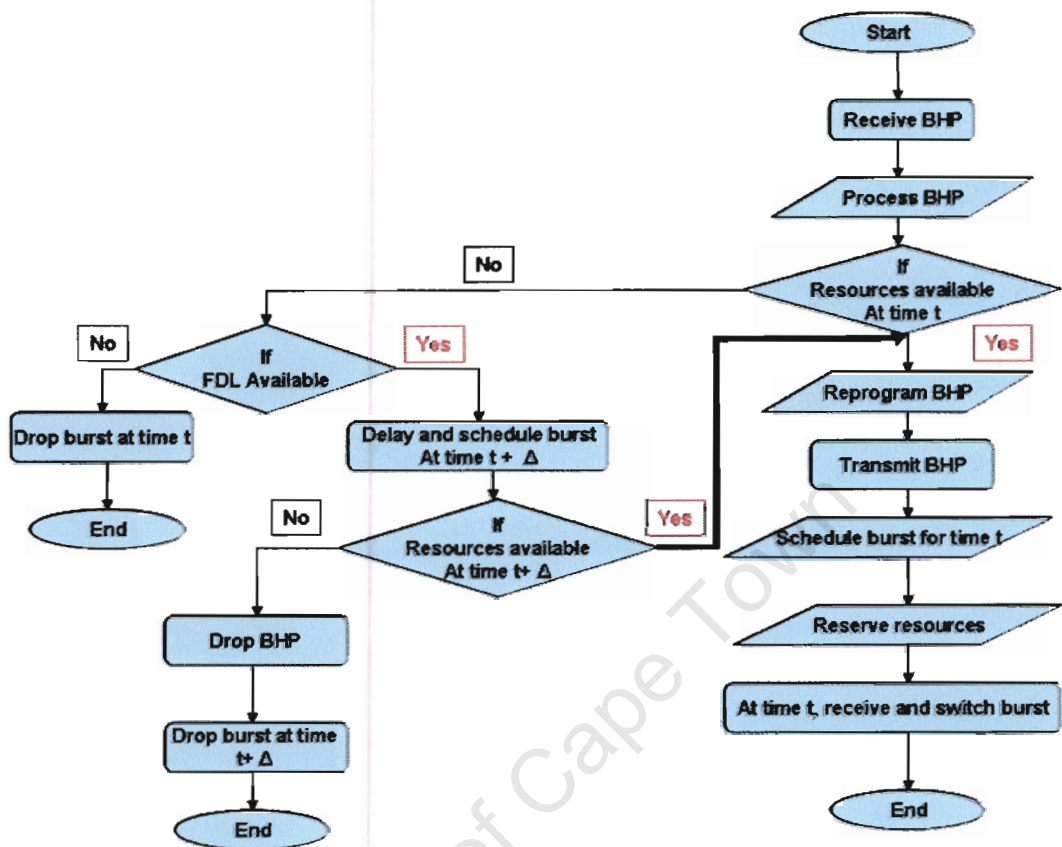


Fig. 2.5 Flow chart of a basic OBS LSR with the option of an FDL.

2.2 Burst assembly

Burst assembly is concerned with the aggregation of packets into bursts. The lengths of the bursts are of critical importance. Long bursts result in increased contention in the core while short bursts create bottlenecks at core routers. There are three basic parameters that must be satisfied by any assembly scheme: minimum and maximum burst lengths, and time or duration of burst assembly. Optimal lengths are critical during assembly, since short bursts result in too many control packets, and excessively long bursts hold core resources for longer times than is necessary, resulting in increased burst contention and therefore more burst drops.

There are generally three methods of burst assembly, from which most schemes may be derived: Threshold based assembly schemes, mixed threshold burst assembly schemes, and adaptive threshold assembly schemes [17 18].

2.2.1 Threshold based assembly

Threshold based schemes can either be time or length based. For time based threshold schemes, bursts are assembled for a fixed amount of time. Bursts therefore arrive at an LSR and egress node (if same route is taken), at a constant inter-arrival time if constant offset times are assigned. The burst lengths will however have variable distributions. For length based threshold schemes, bursts will have almost constant lengths and a variable burst inter-arrival distribution at the LSRs and egress node.

2.2.2 Mixed threshold assembly

Mixed threshold based schemes combine both time and length based schemes to achieve better OBS network performance in regards to delay. Thresholds of both time and length are set, and when either of them is reached, the burst is transmitted. The threshold times or lengths can be varied to achieve optimum performance for particular class types.

2.2.3 Adaptive based assembly

Adaptive based schemes change threshold times and lengths based on traffic characteristics. Traffic modelling therefore becomes essential to the performance of adaptive schemes.

The rapid increase of Internet traffic to-date exceeds traffic due to voice, with voice now being transmitted over the Internet, Internet traffic continues to increase. The concept of burst assembly is not an entirely new one. Voice packets were proposed for assembly in electronic networks, in a bid to improve bandwidth efficiency in the core network. Some electronic burst assembly schemes include TAG. The challenge in burst assembly is in traffic shaping.

2.3 OBS traffic shaping

Several aspects, such as traffic distribution and delay tolerances, of incoming traffic must be considered when assembling packets into bursts. While in the past traffic has been exponentially distributed, works from recent years [19] and [20] have shown a very significant deviation from this norm. Traffic has been shown to be self-similar in nature, with the Poisson model greatly underestimating the burstiness of Internet traffic. The deviation has in part been attributed to the change in the nature of traffic being transmitted in recent years. In [21], it is

further proved that the Poisson model does not represent the burstiness that is observed in Internet traffic today. How the self-similarity affects the performance of an OBS network is still an open question.

However, from past works on Poisson modelled traffic, we can make several assumptions about the expected performance of Self-Similar traffic. With exponential traffic, Jackson queue theory of same distribution applies [22]. It states that if input traffic distribution is Poisson, then the output traffic flows will be of a Poisson distribution too. This assumption has been used extensively in traffic engineering in past years while communication traffic clearly followed the Poisson model, with much success. We may then infer from the same distribution theory rule that:

For an optical Burst Switched Network, if incoming packets to a packet buffer are of a self similar distribution, and are assembled into bursts at the rate that they arrive, then the assembled bursts will be of a self similar distribution as well.

We intuitively state the above rule, and previous results from [23] and [24] do show that assembled bursts do indeed result in self-similar traffic in the OBS network. We however make the assumption that assembler assembles at a rate that limits packet drops due to buffer overflow. This translates to the assembler burstifying packets at a rate more or less equivalent to the rate of incoming packets. Should the assembler rate of burstification be significantly lower than the rate at which the packets arrive, the distribution of the core traffic will be less bursty. Should the assembler rate be higher than the rate of packet arrival, core traffic may indeed be more bursty than the incoming packets. However, this would be an unlikely scenario, since buffer management is a key problem for self-similar traffic.

It should be noted that there are publications that do report that assembled traffic loses its self-similarity [25]. These results can be attributed to the way the assembler is set. Results from [25] do not state how the assembler is configured, which is key to traffic shaping, nor do they specify the behaviour of the packet buffer. The above rule is therefore key to appropriate traffic shaping.

2.4 Contention resolution and burst scheduling

In this section we introduce the fundamentals of contention resolution and burst scheduling. Contention resolution in the core is key to QoS of the bursts and performance of the OBS network. Scheduling schemes are designed to minimize contention of resources while maximizing the performance of the OBS network and maintaining QoS. At the arrival of the BHP, the LSR must select a data channel that is not reserved. Appropriate scheduling of bursts maximizes performance of the OBS network by minimizing voids in the core network to increase channel utilization, and resolve contention. There are three basic methods of contention resolution: by using FDLs, wavelength converters, or deflection routing. Alternatively, a combination of any of the three would result in better overall contention resolution, though that would also increase the cost of the LSRs.

In this section we use the notation nB_λ , where n is the number of bursts, B is the traffic type and λ indicates the wavelength being requested by the burst. Consider an idealistic traffic stream in Fig 2.6 of bursts arriving at an LSR that results in no resource contention. Contention resolution looks to achieve such a scenario, where all bursts have their own period of service. Fig 2.6 is best achieved using Time division Multiplexing (TDM). However OBS is not synchronous, but rather asynchronous.

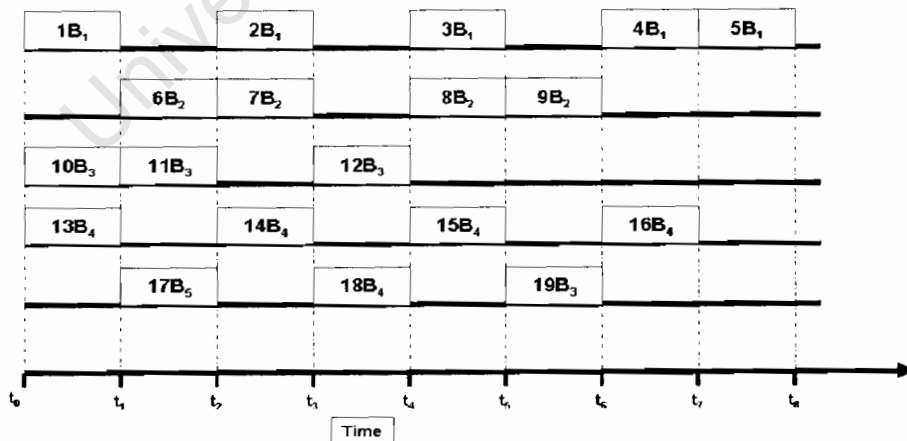


Fig. 2.6 An idealistic burst arrival at an LSR with no resource contention, best achieved in Synchronous networks.

Compare Fig 2.6 idealistic burst arrivals with a realistic burst arrival shown in Fig 2.7. Fig 2.7 shows bursts that contend for resources. There is contention between bursts $1B_2$ and $13B_2$

for wavelength 2 in the service period t_0 - t_1 , bursts $6B_4$, $11B_4$, and $17B_4$ for wavelength 4 in time period t_1 - t_2 , and bursts $3B_2$, $15B_2$ for wavelength 2 in the time period t_4 - t_5 . Contention resolution methods seek to either change the wavelength using wavelength converters or/and service period using FDLs.

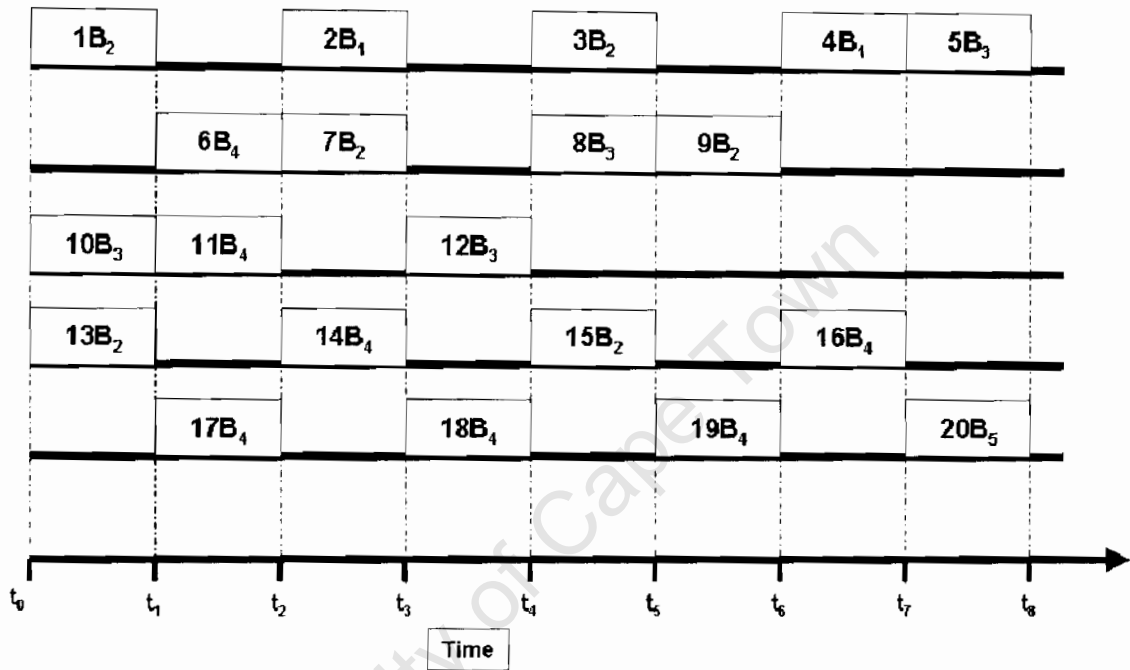


Fig. 2.7 Resource contention of bursts with bandwidth not being fully utilized.

- Using FDLs [26]

With two or more bursts contending for the same wavelength on the same port, the contending burst is delayed in the FDL for the duration of the contention period. Do note that the BHP offset information in the BHP must be updated with the new offset period which will be the length of the FDL used. Fig 2.8 shows how contention resolution is achieved using FDLs. Before contention resolution, in the respective time periods, bursts $1B_2$ and $13B_2$ are contending for the same wavelength 2, bursts $17B_2$ and $6B_4$ are in contention, and bursts $15B_2$ and $3B_2$ are in contention as well.

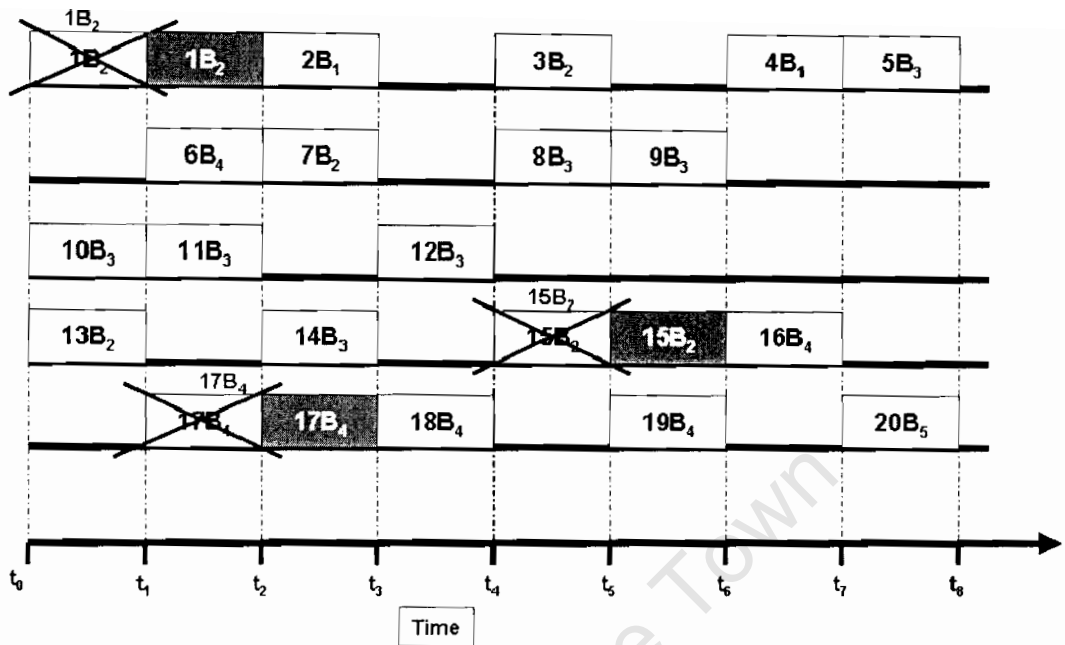


Fig. 2.8 Contention resolution using FDLs. We assume no wavelength converters are available in this case.

Assuming that the FDL on the LSR shown has a length of one burst, Fig 2.8, shows contention resolution of bursts $1B_2$, from time period t_0-t_1 to t_1-t_2 , $15B_2$, from time period t_4-t_5 to t_6-t_7 , and $17B_4$ time period t_1-t_2 to t_2-t_3 .

- Using Wavelength converters [27]

Instead of using FDLs, wavelength converters are normally preferred because it is a more mature technology and will probably be a cheaper option in the future. When two or more bursts contend for the same wavelength, the contending burst is moved to another wavelength that is free in the same time slot. The wavelength field of the BHP is updated with the new wavelength before it is re-transmitted. Fig 2.9 shows how contention resolution is achieved using wavelength converters. Contention is resolved by burst $1B_2$ becoming $1B_4$ (wavelength 2 to 4), burst $17B_4$ becoming $17B_2$ (wavelength 4 to 2) and burst $15B_2$ becoming $15B_4$ (wavelength 2 to 4).

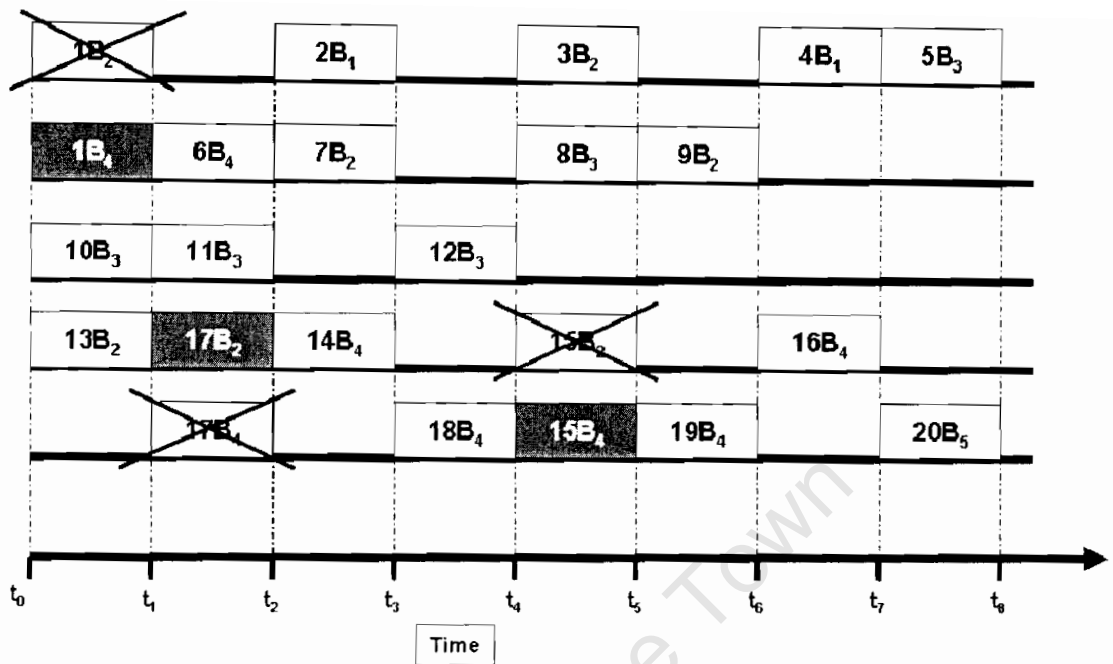


Fig. 2.9 Contention resolution using wavelength converters only.

- Using Deflection routing [28]

Deflection routing can be used with wavelength converters or/and FDLs to achieve more contention resolution. During contention for the same port, a burst may be rerouted through a different port from the original route to an alternative one. Yet the alternative route may be the longer route, and may cause more contention in subsequent nodes. Deflection routing may also result in increased end-to-end delay.

Scheduling of bursts greatly varies with the presence or absence of FDLs and wavelength conversion capabilities. In this study we only consider scheduling schemes without the use of FDLs. Scheduling schemes can be classified as those that allow for void filling and those that do not.

Consider a scenario in Fig 2.10 with Burst B to be scheduled by the scheduler. The symbol λ in Figs 2.10 to 2.14 indicates the wavelength assigned for bursts on that line.

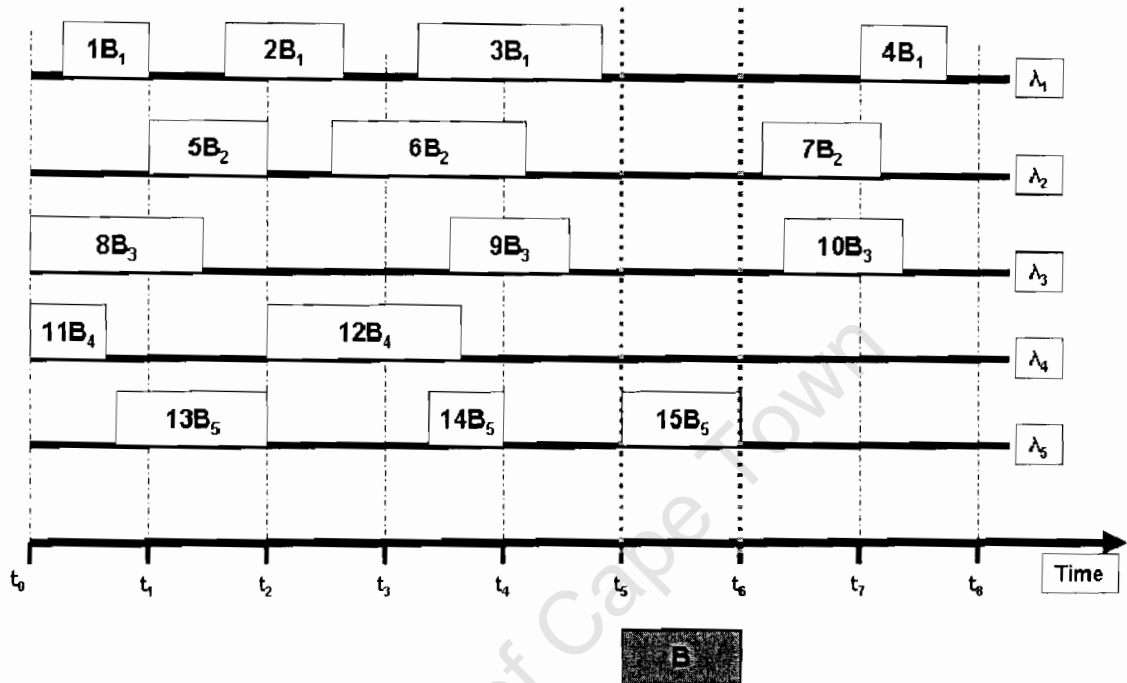


Fig. 2.10 A burst B requiring scheduling.

1. Scheduling without void filling

A non void filling scheme, Latest Available Unscheduled Channel (LAUC) was proposed in [29]. In LAUC, all future time reservation of data channels is maintained at the LSRs by the scheduler. In the core node, this time is the duration of the offset time plus the duration of the burst, only if no FDLs are used. If FDLs are used, then the possible scheduling time increases by the length of the FDL. LAUC seeks to minimize the voids between bursts by allocating the next burst the latest unscheduled channel. Fig 2.11 shows how LAUC works. In Fig 2.11, wavelength 4 at the time slot of the burst arrival t_5 - t_6 , is assigned the burst. Though there are choices in the same time period on wavelength 1 and wavelength 3, the assigned wavelength is one that has the longest unscheduled period.

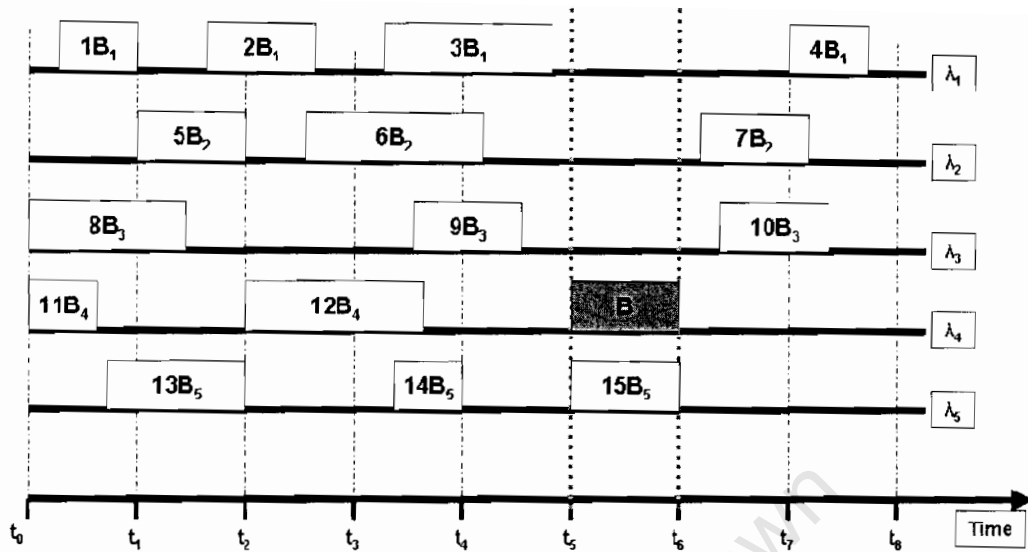


Fig. 2.11 Burst B being scheduled using LAUC without void filling.

Using the example in Fig 2.10, consider a burst arriving at time period t_5 - t_6 to an LSR with several data channels for a burst B. The algorithm chooses a channel, if free, with the latest available unscheduled wavelength, and has the smallest gap between the previously scheduled burst.

Horizon is proposed in [30]. Though proposed independently Horizon works in the same way as LAUC. The scheme schedules bursts on a wavelength with the longest Horizon. Where the horizon time is the time after which no reservation is made on a given channel. The longest Horizon is the same as the Latest Available Unused Channel (LAUC). Therefore Fig 2.11 also illustrates how Horizon operates.

LAUC and Horizon are simple algorithms, and have good performance in terms of speed. However they result in lower utilization of wavelengths because no void-filling is done. The void filling scheduling schemes have more efficient use of wavelengths.

2. Scheduling with void filling

Latest Available Unused Channel with Void Filling (LAUC-VF), also proposed in [29], improves LAUC (and Horizon) and proposes the filling of voids between bursts. The LAUC-VF scheduler monitors the void intervals between bursts. The scheduler fills the void with enough gap for the burst being scheduled, and with the latest starting void Fig 2.12. LAUC-VF results in less burst loss compared to Horizon, but is significantly slower. However, in [31] a Minimum starting void (Min-SV) algorithm is proposed. The

scheme achieves the same objective as LAUC-VF. The difference is in the implementation. For Min-SV, a balanced binary search tree is used to determine the minimum starting void. As a result, performance of Min-SV achieves the same low burst loss compared with Horizon, while achieving the performance speed of Horizon. Minimum Ending Void (Min-EV) [31] is the opposite of Min-SV by minimizing the end of the void filling burst and the end of the void Fig 2.13. Best-fit can be seen as a compromise between Min-Sv and Min-EV. It schedules a burst into a viable void that leave the minimum void on either end of the void filling burst Fig 2.14.

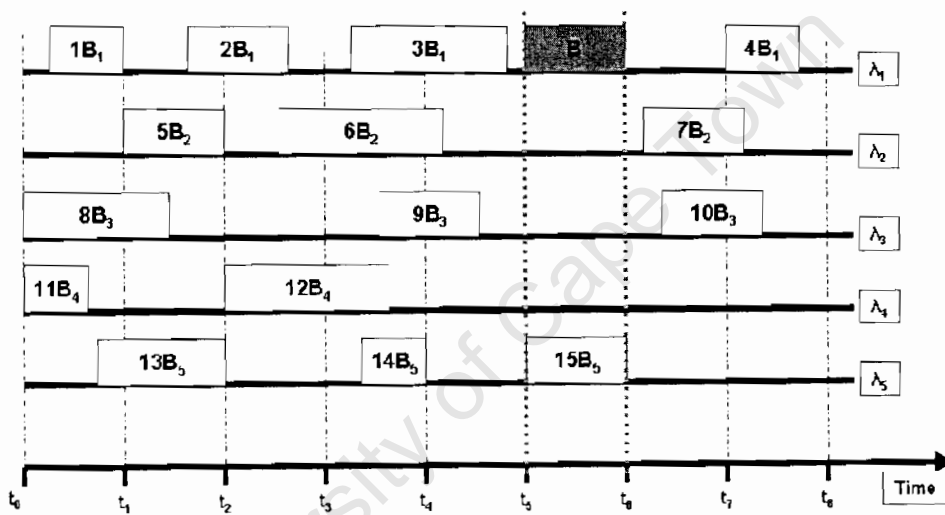


Fig. 2.12 LAUC-VF scheduling.

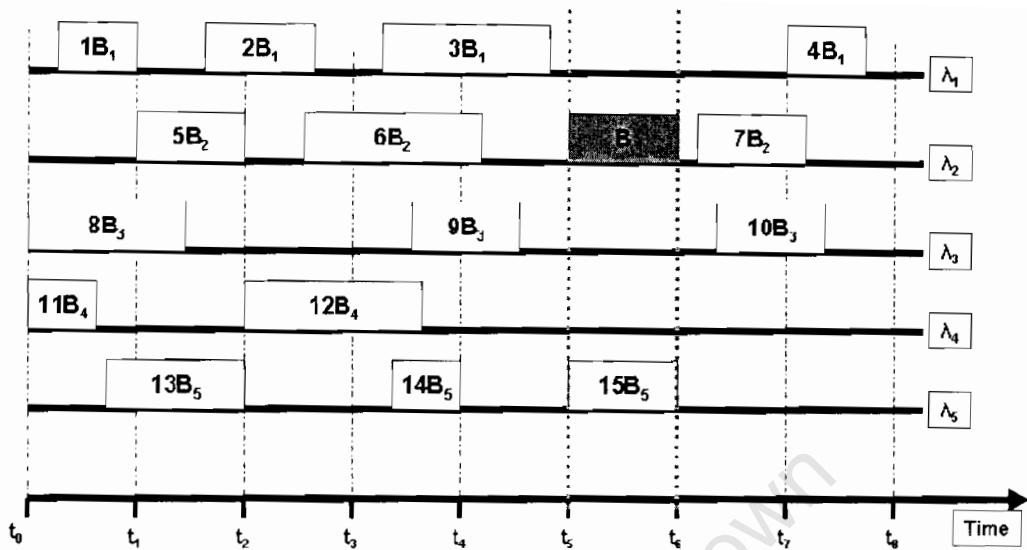


Fig. 2.13 Scheduling with Min-ev.

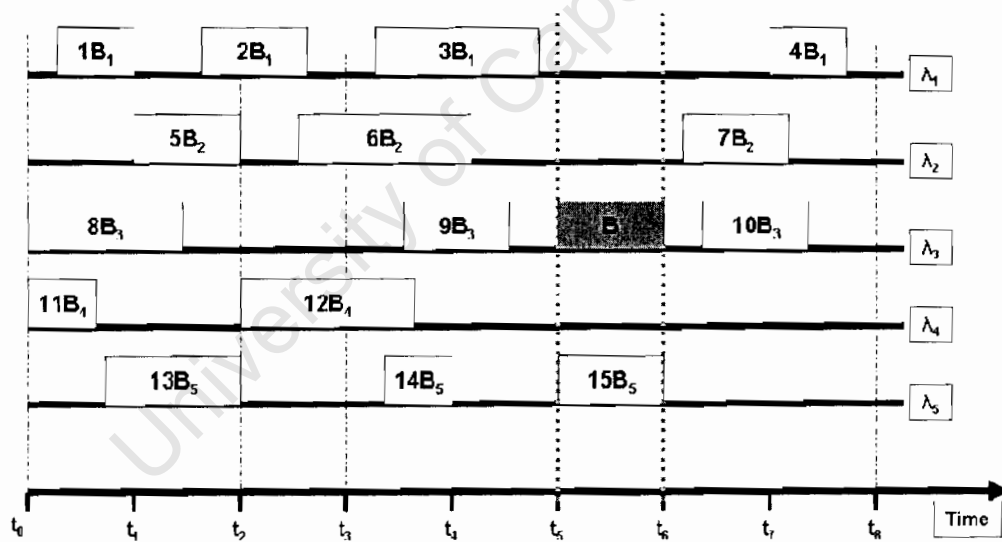


Fig. 2.14 Scheduling using the Best-fit method.

Recent proposals have focused more on scheduling of bursts based on the arrival of the bursts themselves, unlike previous schemes that schedule bursts solely based on the arrival of the BHP. Scheduling bursts based on their arrival times at an LSR introduces a new focus of buffering BHP packets.

One such proposal made in [32] is Pipe Line System. The proposed Pipe Line System seeks to schedule a burst using LAUC-VF while maintaining a FIFO of the bursts. The need for this scheme as is due to the fact that for variable offset times, the arrival times of the BHPs do not correspond to the arrival of the bursts. Thus, by buffering of the BHPs in a buffer and retransmitting them according to the order of the bursts' arrival does minimize voids.

Another such scheme is proposed in [33], is Virtual Fixed Offset time (VFO). In VFO bursts are assigned variable offset times but interestingly models a constant offset time. Bursts are processed in order of arrival of the bursts rather than the arrival of the BHP packets. In VFO, when a BHP is received at the LSR, it does not immediately schedule the corresponding burst. The BHP's offset times are compared, and bursts are scheduled according to the burst arrival. However, there is a need to electronically buffer the BHPs. Buffering of the BHPs is a result of having less control channels than data channels. Buffering of BHP packets results in a decrease of the offset time. It then becomes necessary to buffer the bursts as well to increase the decreased offset time.

The disadvantage in these schemes is the cost of introducing of FDLs at all the LSR in the OBS network. More intelligence is also required in the LSR nodes too, which undermines one of the key advantages of the OBS technology for a simple and cheap OBS core network.

2.5 Resource reservation

Reservation schemes are mainly differentiated by the time of reservation of resources from when they are requested (scheduled) to when they are released. Reservation of resources can either be immediate or delayed. The release of resources can either be timed for the duration of the burst or be explicit, using a release packet after the burst.

There are four variants of OBS reservation schemes. These are Tell and Go (TAG), Tell and Wait (TAW), Just in Time (JIT), and Just Enough Time (JET). In this section we detail all four schemes.

2.5.1 Tell And Go (TAG)

In [34, 35], TAG is proposed. In TAG, the burst is transmitted directly after the BHP assuming a negligible processing time for the control packet. An LER transmits the BHP to set up a path for the burst. The BHP is still in the control channel and the burst in the data channels. Because the scheme does not take into account the processing realistically, buffers must be used at each of the LSRs. Due to the need of long FDLs, the TAG reservation scheme is not suitable for an ideal OBS. The FDLs would make an OBS network very expensive.

2.5.2 Tell And Wait (TAW)

In TAW, unlike TAG, the BHP is transmitted from the ingress node to the egress node while the burst remains at the ingress node. The BHP is processed at each node, and an acknowledgement sent to the ingress node from each node. Upon receiving all the acknowledgements of a successful reservation, the burst is then transmitted. TAW eliminates the need of FDLs, unlike TAG and would therefore result in a cheaper OBS network. Though a light path is not physically set up from the ingress to the egress of the network, the delay incurred by TAW is comparable to that of circuit switching. LSR resources are therefore held for long periods of time. By attempting to avoid the use of FDLs, a compromise is made on delay. This would still not be acceptable for NGN networks.

2.5.3 Just In Time (JIT)

In [36] JIT is proposed. In JIT the BHP is transmitted with an offset time with respect to the burst. At an LSR, the BHP requests and reserves (if available) a data channel. The data channel is then reserved from the moment of reservation to the expected end of the burst. JIT therefore reserves a data channel for the duration of the remaining offset time, during which it remains idle. JIT is an improvement on TAW in terms of delay incurred by the bursts and duration of resource reservation. The JIT protocol is designed for DWDM technology, and for low latency for unidirectional transport of bursts. Fig 2.15 shows how JIT works.

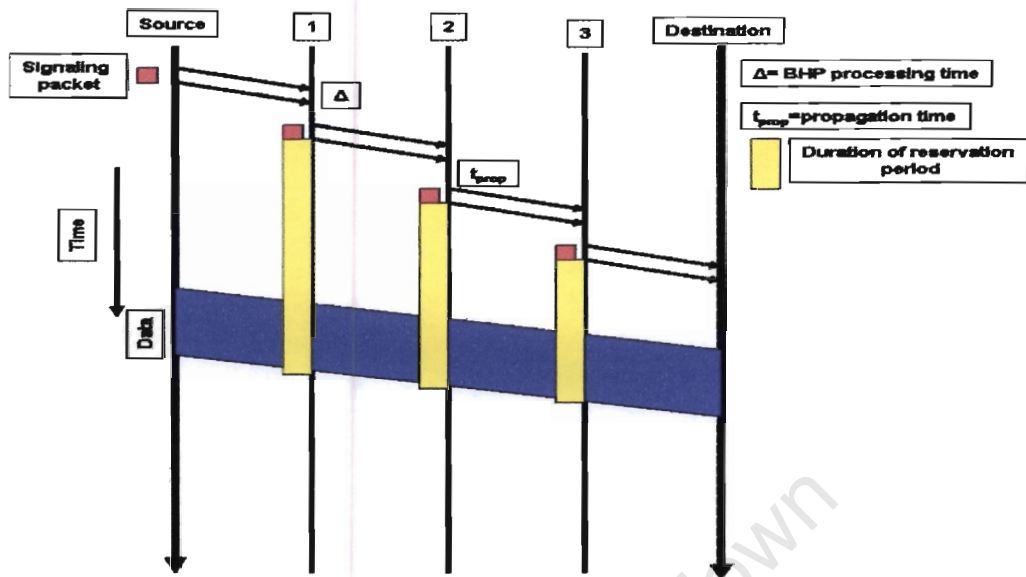


Fig. 2.15 Performance of JIT.

Fig 2.16 shows a method that may be employed to more accurately predict the end of a reservation period. Fig 2.16 includes a release packet after the burst to explicitly release the reserved resources. The release packet may however increase contention in the core network.

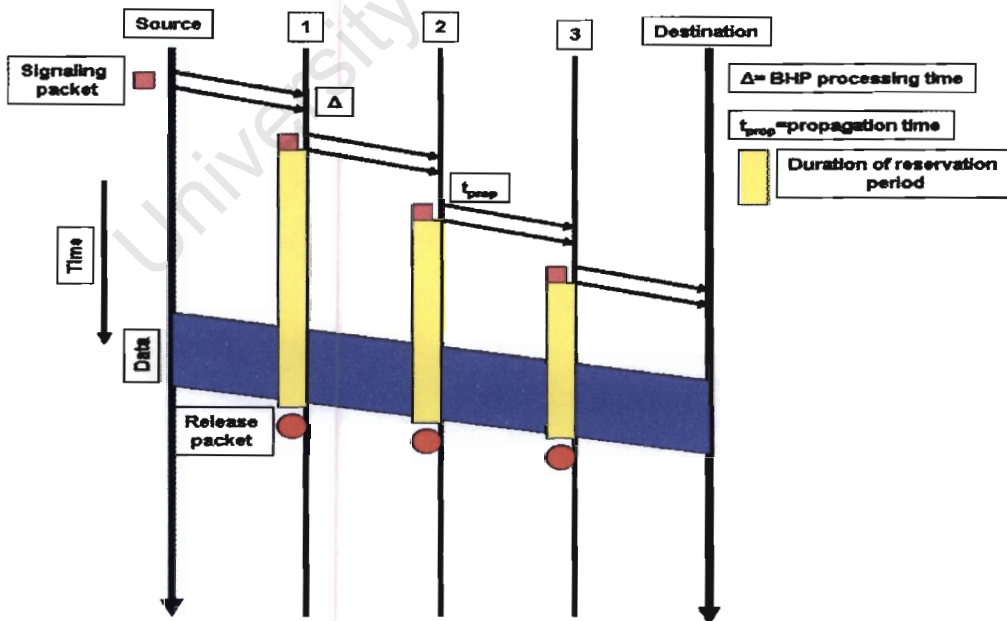


Fig. 2.16 JIT with release packets for each burst.

2.5.4 Just Enough Time (JET)

JET is proposed in [38] based on TAG and improves time of resource reservation in JIT. In JET unlike JIT, reservation is done from the expected time of arrival of the burst, to the expected end of the burst, that is, only for the duration of the burst. This is called delayed reservation. This method reduces contention because of the reduced reservation period. Void-filling is therefore possible with JET, since reservation does not only start at the expected time of arrival of the burst. Fig 2.17 shows how JET works. The delayed reservation feature in JET allows for a more efficient reservation scheme for bursty traffic of which OBS traffic is. In case of contention, a burst would be dropped. Contention resolution is also easier due to the shorter reservation periods. However, if FDLs are used, JET allows for the burst to be delayed for the duration of only the burst length.

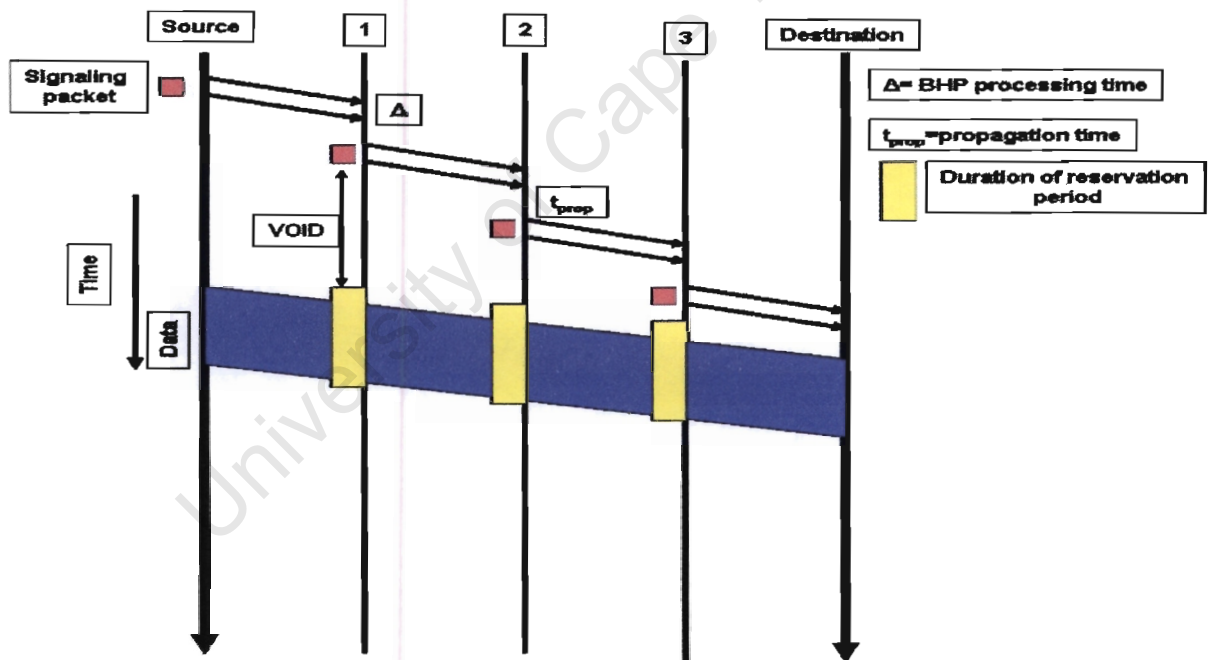


Fig. 2.17 Performance of JET.

The JET reservation scheme can be used for offset QoS differentiation of bursts. In offset-based QoS schemes, the higher the priority, the longer the offset. Therefore, bursts with higher priorities may be given priority over lower priority bursts should there be contention of resources. However QoS differentiation in OBS is not that straightforward, since extra offset increases the

end-to-end delay of bursts. Several other factors must be considered, which we do in our proposal.

2.6 Burst segmentation

In this section we review a variant of OBS proposed in [38], Time Sliced Optical Burst Switching (TSOBS). TSOBS is synchronous in nature, in contrast to OBS which is asynchronous. In TSOBS, bursts are assembled in fixed time slots. A burst is therefore made up of time slots of fixed sizes. During assembly individual frames are filled up with packets and the frame added to the burst. If the frame is not full at the end of the assembly time, the frame is padded to fill the empty space. The lengths of the bursts are therefore counted in terms of frames. Several time slots make up a frame, with each of the time slots having a fixed size. Fig 2.19 shows how a TSOBS burst may look like.

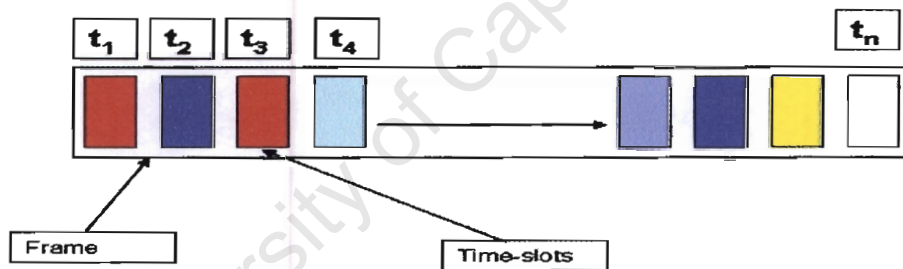


Fig. 2.18 A TSOBS burst.

In TSOBS, switching is instead done in the time domain instead on the wavelength domain. Switching in the time domain entails buffering of the burst so as to schedule the burst. The authors in [38] justify the choice for using optical buffers by showing that the amount of storage that would be needed is less than 1% in comparison to the amount of storage needed for packet switching, and therefore might actually be a cheaper alternative since the cost would be greatly reduced. TSOBS compares to SONET/SDH by using data being carried on wavelengths in a Time-Division Multiplexed (TDM) manner.

Perhaps one of the notable differences between TSOBS and OBS, is the amount of processing required at each of the LSRs. To facilitate this processing, a major component of a

TSOBS router is an Optical Time Slot Interchanger (OTSI). Using the OTSI, each of the time slots in a frame can be aggregate with another to reduce processing overhead. OTSI also have the task of switching them to their appropriate destinations. Switching in the time domain as opposed to switching in the wavelength at the same time maintaining the data burst in the optical domain demands that buffers be used. The authors propose for each time slot a fixed one-microsecond slot. With a 40Gb/s transmission speed, 4400 bytes of data would be fit in each time slot, with a frame capable of having 350 timeslots. A frame therefore has the capacity to carry a lot of user data compared to OBS, where it is proposed that a burst have a maximum of 40000 bits. The rest of the TSOBS works the same way as OBS. The objective of assembling of bursts into frames is in the ability to resolve contention. Each Frame can split into its individual time slots at the LSR.

All burst assembly and reservation schemes will be impacted by the nature of traffic that is being transported. With Internet traffic having been shown to be bursty and self similar, we investigate its impact on an OBS network. For definition and generation of Self-similar traffic see Appendix B.

Chapter 3

3 Previous Assembly Schemes

In this chapter, we discuss works that have laid the foundation using previous assembly and burst scheduling schemes at the edge node.

3.1 Assembly of Packets to Bursts

It is common for many assembly schemes to assume that due to the bufferless nature of OBS queuing theory falls away. This assumption is valid for the LSR nodes, if no FDLs are used. Queuing theory though does come into play when designing assembly schemes at the LER nodes. A question that must be addressed is whether the problem of burst assembly is a queuing theory problem requiring a queuing theory answer. A QoS guarantee for different classes of traffic in the core is still an open question, and is addressed in this chapter.

3.2 General assembly schemes

Burst assembly schemes are generally either timer-based or threshold based. In timer-based schemes, bursts are assembled and transmitted at periodic time intervals. The variance of traffic at the ingress node results in bursts being of different lengths in the core network. In threshold based schemes, the number of packets in a burst is limited to both maximum and minimum number of packets.

Both the timer and threshold based schemes have different objectives. The timer-based schemes are primarily concerned with limiting the delay incurred by the packets. Therefore, bursts with a delay tolerance t_d that are destined to a given egress node D whose shortest propagation period t_{prop} will be transmitted at time intervals t . It is obvious that due to the variance of the traffic being assembled, there will be bursts that are very short, and those that will be very long, depending on the amount of traffic available, which will greatly affect the performance of the OBS network. The threshold based scheme on the other hand is primarily

concerned with the efficiency of the OBS network. Due to the variance of the assembled traffic bursts' lengths are limited to within the maximum and minimum burst lengths. However, this method has the disadvantage of not taking into account the delays incurred by the packets being assembled. There is clearly a trade-off between the performance of the OBS network, and delays incurred by the bursts and the two schemes actually complement each other.

3.3 Assembly with QoS specifications

QoS is mainly determined by time spent by packets at both the LER and in the core, and burst loss. While most studies mostly consider packet loss ratio as the main contributing factor, the residence period in the OBS network, especially at the LER, of the packets is rarely taken into account. One such study is done in [39].

In [39], a scheme that should guarantee end-to-end QoS is proposed for a bufferless OBS network. Two classes of traffic are considered, real time and non-real time traffic. Real time traffic is given the higher priority over the non-real time traffic. While there is at least one wavelength available, no pre-emption occurs, and all wavelengths are shared according to the demand from each class. In a case when a real time traffic burst arrives for transmission while a non-real time traffic burst is being transmitted, pre-emption occurs, and the real traffic burst takes over the wavelength. To maintain QoS, a packet loss ratio (PLR) is set to avoid excessive drop of either class above which QoS would be compromised.

In [40] a mathematical analysis and simulation of self similar traffic over an OBS network is done. The study concludes mathematically and by simulation, that self similarity would have an adverse effect on the buffers in the LERs and the LSRs. Two assembly parameters are introduced in [40]: a Time Out Factor (TOF) and a Burst Size Factor (BSF). Both factors are ratios, with TOF being a ratio of the time it takes for a packet to be added to a burst to the inter-arrival times of each FEC, while BSF is the ratio between a set fixed burst size of a burst for a given FEC, to the maximum IP packet size. Reasons for the choice in parameters are not clearly outlined, however, the ratios of both parameters can intuitively be understood to vary with change in self similarity and rate of burstification. The study shows that there may be optimal TOF and BSF ratios for minimum loss probabilities at the LER.

The treatment of self similar traffic in [40] follows from proposals from [20], which are widely accepted, and used in this study. A FEC is also taken into account, as would be expected in an assembly scheme. The scheme does not however address offset assignment or/and treatment of the burst queues, which would have a direct impact on the BSF.

In [41] an assembly scheme is implemented on an Intel IXP2400 network processor. In the processor, a Basic Assembly Unit (BAU) is implemented, one that uses the mixed timer burst length assembly scheme. The contribution in [41] is mainly the implementation of OBS on a processor, showing that OBS is a realizable protocol even with processors available to-date (2006). Because implementation of an assembly scheme dictated a realistic LER set-up, two queues are used; and input and output queue. However no attempt was made in modelling of traffic into the packet queues, nor was offset assignment considered in this work. By providing QoS independent of the delay tolerance of individual packets and bursts, bursts and packets that should not be dropped are more likely to be dropped.

Timer based algorithms are studied in [42]. An Adaptive-Assembly-Period (AAP) algorithm, derived from a Fixed Assembly Period (FAP) assembly scheme is proposed. They show that by assembling bursts using a fixed time (FAP), performance of the network is greatly reduced. They then attribute the lack of performance to the lack of flexibility in the assembly scheme, hence the algorithm is adaptive. The algorithm adapts to the changes in variance in the incoming traffic by changing the assembly periods of the bursts.

In [42] before burst assembly, packets are buffered into queues, according to their destination. At the front of each queue before the burstification of the next burst, the length of the previous burst is noted. Depending on the previous burst length the length of the current burst is adjusted to meet network performance. Results from the study in [42] show that the AAP performs better than the FAP, and the mixed-timer-length algorithms.

The results presented are useful in understanding the impact of traffic variance on burst assembly. However, the extent to which the modelled traffic in [42] models realistic traffic is not clear. They use TCP/IP traffic for incoming traffic into the ingress node. It is not mentioned in

the paper as to whether the mentioned variance therein matches the self-similar traffic observed in today's Internet traffic.

The results do however give insight on how TCP reacts to assembly schemes. They conclude that the AP scheme does not perform well and any assembly scheme should adapt to TCP sending window size. In this study we do not use TCP/IP traffic at the ingress node, but use UDP traffic. Though in our considerations we do not use TCP, one of the QoS parameters that we consider is delay tolerance. Therefore, should traffic input be TCP, our scheme should be able to serve TCP based traffic as well by using the required TCP window as the delay tolerance of the packets.

One major assumption made though is that by the time a burst is assembled, the BHP has already set an appropriate offset and the burst therefore needs no buffering before transmission at the ingress node. Setting of offset and burst buffering greatly affects the performance of the OBS network.

In [43] a Composite Class Burst (CCB) assembly scheme that takes into account delay tolerance is proposed. CCB defines burst classes with respect to different assembly times after which a burst is transmitted. Assembly of bursts using CCB allows for different classes packets to be assembled in the same burst. By assembling different packet classes into a burst while taking into account the QoS demands of all packets, this scheme offers a good method of assembly. However, no clear mention of how offset assignment is to be done is mentioned. It is assumed in this work that extra offset assignment results in poor QoS and classifying of bursts eliminates the need for extra offset assignment schemes. While it is true that assigning classes to bursts does improve QoS differentiation in the core, the assumption that extra offset assignment always results in poor QoS is not justifiable.

An extra offset time QoS scheme is proposed in [44] using FDLs and in [45] without FDLs. Extra offset assignment schemes rely on the fact that delay tolerances of bursts are significantly longer than the time period required to route a burst from the ingress node to the egress node. The idea of assigning extra offset times to bursts to reduce contention in the core is therefore a credible one. There is a limit to how much extra offset can be assigned without

violating QoS demands of the burst. Burst classes, as in [43] must therefore be considered. In [46] two classes are considered, real time and non-real time. It is shown that by assigning longer offset times to real time applications their probability of blocking reduces significantly. The result in [46] does also imply that extra offset assignment can be set to not fall out of range of the delay tolerances of bursts delay tolerances. In [47] it is argued that assigning long offset times to high priority bursts results in prolonged end-to-end delays. This is true, though a re-definition of long offset time should be limited to fall within the end-to-end delay tolerances of a given burst class.

In [48] a threshold assembly scheme is proposed instead, and contention resolution done using priority based segmentation. Segmentation policies demand that each field of a burst have a specified length, and burst classes have fixed lengths. This method would result in long queues at the ingress nodes due to the slow generation and transmission of bursts. This scheme would especially not perform well with bursty traffic.

Bursty traffic has been taken into account in some proposals. In [49], where QoS is primarily based on burst drops, burst length distribution rather than offset is used as a QoS factor. It is shown in [49] that Pareto distributed burst lengths do result in low burst drop probability for high priority bursts and high burst drop probability for a hyper-exponential burst length distribution. It should however be evident that by manually dictating the distribution of bursts, delay of packets and bursts at the LER is completely ignored. Shape of OBS traffic cannot be predetermined just as Internet traffic cannot also be predetermined. In our work, we let delay constraints of Internet traffic to dictate burst length distributions.

All proposals seem to have made a contribution in one aspect and making several assumptions for others. We seek to make a proposal that takes into account most important OBS network aspects so as to make our proposal as realistic as possible.

Chapter 4

4 Proposed Assembly and Offset Assignment Scheme

In this chapter, we propose an FEC-Based assembly algorithm and a Pareto-offset assignment scheme as a complementary procedure to the proposed assembly scheme.

Most assembly schemes have not taken into account the realistic burstiness of the incoming traffic and the effect on the QoS of the assembled bursts, and for this reason cannot realistically meet traffic QoS requirements. We aim to design an optimal burst assembly and offset assignment scheme for OBS networks using queues, for the arriving packets and for the departing bursts at the LER. When assembling traffic, with QoS in mind, we argue that it is imperative that offset assignment be done as a complementary procedure to burst assembly.

During the burst assembly, we make several considerations for both the packet and burst queues. From the given considerations, we can then optimally design an FEC and queuing scheme that will satisfy QoS demands at the LER. We do not derive a mathematical analysis of the GI/G/1/m queue, but only implement the queue.

Key to this proposal is the modelling of the incoming traffic at the ingress node. We consider self-similar traffic from multiple sources, each packet with a given delay constraint. Unlike previous schemes, this scheme aims to satisfy the delay constraints of packets, at the same time minimize contention.

Contributions in this chapter are:

1. An abstract traffic model of OBS traffic is proposed and QoS considerations for OBS networks outlined
2. A QoS assembly scheme at the LER using decision theory
3. An offset assignment scheme at the LER. The Offset assignment is random with a Pareto distribution.

In the next section (4.1) we propose a method offset assignment and a mathematical model of OBS traffic. We also outline what we consider to be realistic considerations for QoS based assembly. The objective of this section is to give guidelines on what parameters to consider if QoS is to be realised. These parameters are used for QoS classification at the LER during

assembly. Section 4.2 then discusses how received packets at the LER are assigned to their respective queues with respect to parameters discussed in section 4.1. The mechanism described section 4.2, of arranging packet queues, determines the rate at which each class is burstified by the assembler. The burstification process is discussed in sections 4.3 and 4.4. Section 4.3 discusses how the proposed four classes of packets are assembled using decision mathematics to maintain QoS. Section 4.4 then discusses the assembler itself. The proposed assembler uses the criteria set for it in section 4.3. From the assembled bursts (section 4.4), a queue of bursts is formed as described in section 4.5. In section 4.5, we set out rules for satisfying the QoS demands of the assembled burst. Section 4.6 discusses the proposed queue network as a whole and a summary of the proposal is given in section 4.7.

4.1 Design Considerations

In this section we describe considerations that ensure QoS in the assembly and offset assignment scheme proposed. We use these considerations to implement a Forward Equivalence Classification (FEC) that can optimally cater for traffic QoS.

4.1.1 Considerations for burst assembly

We identify the following factors to be critical to the optimal performance of an OBS network: modelling of OBS traffic, destination of the burst, burstiness of incoming traffic, delay tolerance restrictions of the received packets, and minimum and maximum burst length restrictions.

- Modelling OBS traffic

The difficulty of modelling self-similar traffic has led to many researchers modelling burstification by assuming a Poisson model. In this section we derive an abstract mathematical analysis of self-similar traffic in the context of OBS traffic. From [25] it is shown that assembled Internet traffic smoothens a little, but self-similarity persists and LRD of the assembled traffic still exists. A comparison between OPS and OBS is made, with self-similar traffic as the incoming traffic of the ingress node. The R/S distributions in [25] show that the distributions of the assembled bursts compared to that of OPS

deviate slightly. The result shows that assembled traffic though smoother, is still significantly bursty and exhibits self-similarity.

One challenge in OBS is in the mathematical treatment of the OBS-traffic. We propose that it is feasible to model OBS-traffic as ON/OFF periods. The burst period preceded by an offset period, gives a natural correspondence to Pareto ON/OFF periods as illustrated in Fig 4.1.

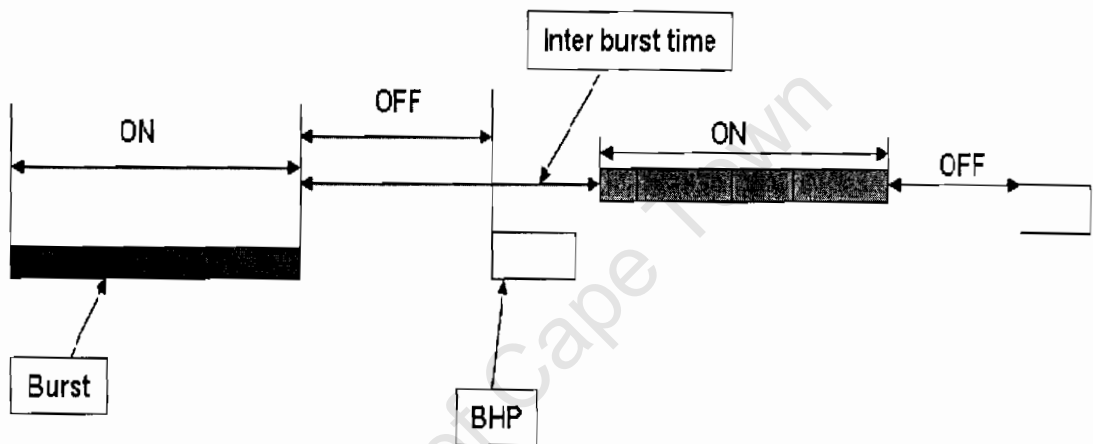


Fig. 4.1 Proposed offset assignment scheme.

ON periods in Pareto would be modelled as the burst duration, while the OFF periods would be modelled as the offset times. The modelling of OBS core traffic in this manner would allow for a per channel flow modelling in DWDM wavelengths. Fig 4.2 shows the overall proposed LER traffic set-up. Modelling OBS traffic as ON/OFF periods in DWDM conforms to the same distribution theory in Jackson queues.

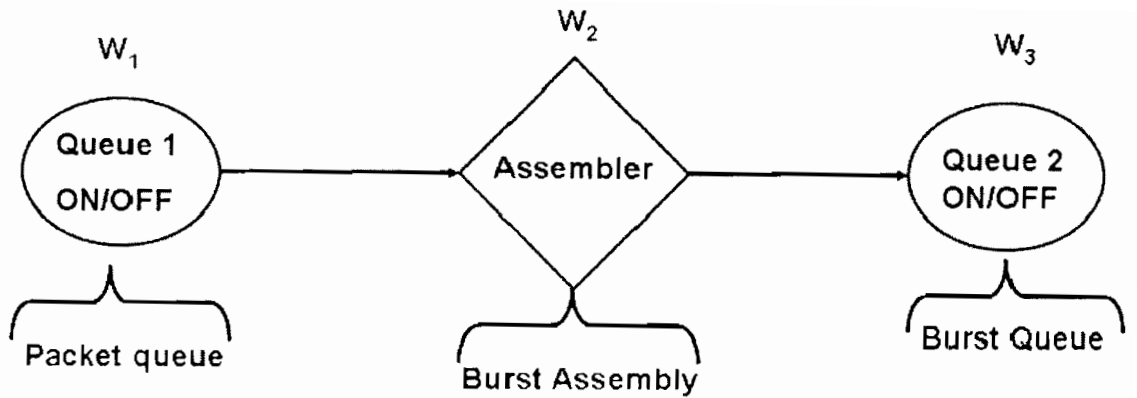


Fig. 4.2 The proposed addition input, output queues and ON/OFF modelling of the traffic.

Consider traffic that has been assembled at the LER, and is being transmitted as per OBS protocol using DWDM with N wavelengths. The total load L at the LER is the sum of the individual wavelength load L_i , where $i \leq N$. For the N sources, therefore,

$$L = \sum_{i=1}^N L_i \quad (1)$$

For every individual wavelength we can find its associated load, with every burst generation and offset assignment. Equation (2) shows load contributed by one burst and offset assignment, and equation (3) shows the total contributed load of the wavelength as a whole:

$$L_i = \frac{ON_i}{ON_i + OFF_i} \quad (2)$$

$$L = \sum_{i=1}^n \frac{ON_i}{ON_i + OFF_i} \quad (3)$$

We can use the burst and offset averages to compute equation (3), for a given channel. However, due to the rapid variance of the bursts, in case of QoS assembly schemes, and random offset assignment schemes, it is more appropriate to have the ON/OFF averages represent class average burst and offset lengths respectively so as not to get a biased mean. Equation (3) would therefore be the average load per wavelength of a given FEC specified class. For a given traffic class, we can therefore estimate the average offset lengths from Equation (3).

$$OFF = ON \left(\frac{1 - L_i}{L_i} \right) \quad (4)$$

Equation (4) allows for a given load, limited to a specified value, while knowing the average burst lengths of that particular traffic class. By determining the average offset times, routing of traffic can be predicted. In line with the definition of self-similarity, both the OFF and ON periods must vary in a Pareto distribution.

- Destination of burst

Packets assembled in a burst must have the same destination. However this condition does not imply that every packet having the same final end user destination must be in the same burst. Assembling bursts by destination ensures that traffic entering the network does exit the network. Assembling by destination is a standard way of assembling bursts but this condition alone is not enough, burstiness of the traffic at different time intervals must be taken into account.

Destination egress nodes may either be near or far from the source ingress node. We classify near destinations as nodes that are near in regards to the number of hops that need to be traversed from the ingress to the egress node. One LSR between the ingress and egress nodes would classify as a near destination, while for example five LSRs between the ingress and egress node route would be a far destination. The second classification of near and far destinations is in relation to the total delay propagation due to the lengths of the fibre links between the ingress and egress nodes. With this classification, burst assembly can be varied in several ways.

When distance is considered in terms of hops, we propose that for long distances, bursts not be long, and for short hop distances, bursts are preferred to be long. Long hop distances imply that the probability of contention at LSRs is increased. Hence the less the number of LSRs the less the total probability of contention. It then follows that for short hop distances, the bursts may on average be long. Higher priorities are then given to short bursts, which would be assumed to be long hop distance bursts, over long bursts that would be assumed to be short hop distance bursts.

When distance is considered in terms of link delay, bursts may either be long or short. Distance by propagation delay has no effect on the length of the bursts but must be taken into account when delay tolerance of the bursts is a constraint.

It is evident that during the assembly, the number of hops that must be crossed should be taken into account to lower the probability of contention, and hence lower the drop probability. This classification, however, does not take into account the burstiness of traffic, and assumes a constant mean. Below we add burstiness to the assembly criteria.

- Burstiness of traffic

In addition to destination of the packets, the burstiness of the incoming traffic is taken into account. For bursty traffic, during periods of prolonged burstiness at the packet queue, assembled bursts to a given destination are more likely to have maximum lengths, while at periods of low burstiness, assembled bursts might have minimum lengths.

Assuming that we know, or limit the amount of traffic of a given class to a load value of L_i , then it would be necessary to determine the expected burstlengths for a given burstiness. We can use the shape parameter α to give an estimate of burstlengths for a given burstiness. We do this by using the first moment of the pdf Pareto distribution see APPENDIX B. The first moment of Equation (3) in APPENDIX B is,

$$E(x) = \frac{\alpha b}{\alpha - 1} \quad (5)$$

Since b is the minimum value that x can take, we have $b = l_{\min}$ the minimum burst length or $b = (off_{\min})$ the minimum offset time. The minimum offset time must be greater than or equal to the minimum time it takes to reach the destination. l_{\min} and off_{\min} are scale parameters.

By considering the burstiness of the incoming traffic, assembly of bursts can be optimized to avoid or limit overflow of the buffers at the ingress node. This introduces buffer management for OBS ingress nodes. With high variance of incoming traffic, the rate of

burst assembly of low delay tolerant traffic must be high, while the rate of assembly of higher delay tolerant traffic would be reduced.

- Delay tolerance of packets in burst

The rate of assembly of each of the classes is dictated by the burstiness of incoming traffic. Packets with equal delay slots tolerances and with the same destination should be assembled into the same burst. Assembling packets by considering delay tolerance and destination results in an assembled burst having different packet types to a destination. By considering delay tolerance, QoS of the packets assembled is provided. The lengths of the bursts are dictated by the delay tolerances of the individual packets.

- Minimum and maximum acceptable burst lengths

When the assembly of a burst starts, its delay tolerance and destinations are read, from which the duration of burst assembly can be determined. There are limitations to the minimum and maximum burst lengths despite the amount of time available for assembly. When a burst is assembled by considering the burstiness of traffic, destination of packets, and delay tolerance, it must not violate the maximum and minimum burst length limits. The minimum burst length limitation may be enforced by padding. By keeping within the limits of maximum and minimum burst lengths, we ensure that the OBS network functions optimally.

Together, these factors should be taken into account during Forward Equivalent Classification FEC of packets and bursts. Packets from various sources going to the same destination are assembled into the same burst. The packets need not be from the same source. However, burst assembly should be complemented by appropriate offset assignment, if end-to-end QoS is to be achieved.

4.1.2 Considerations for setting offset times

An offset for a burst is determined during the assembly of the burst. The alternative to this would be to determine the offset after the burst is assembled. The advantage of assigning offset times during the assembly period is that, should a packet of low delay tolerance arrive in a

burst then immediately after that, a lower delay tolerance packet arrives in the same category of burst assembly requesting for a shorter offset time, the assigned offset time must be re-computed to accommodate the new packet arrival. If during the assembly, the delay tolerance equals the offset time before the minimum burst lengths requirements are reached, the assembler must pad the remainder of the burst with higher delay tolerant packets in the queue. We make the following considerations when setting offset times for each of the bursts being assembled: Priority assigned to an assembled burst, overall delay tolerance of the burst, destination of the burst and possible routes that may be taken from the LER.

- Priority

Priority based offset assignment is considered in this study. Priority of a burst is dictated by the delay tolerance of a burst. Without consideration for delay tolerance, high priority bursts should have long offset times, and low priority bursts have short offset times. Short offset times however limit the ability of a scheduling protocol from adequately scheduling traffic at high traffic loads. In contrast, long offset times would allow for more flexibility. We let the long offset times to be long and random instead of static.

- Delay tolerance (QoS)

Bursts being transported must reach their destination within the bursts' delay tolerance time interval. Ensuring that bursts are not dropped during routing does not necessarily guarantee QoS. Offset times then need to be set so as to not exceed the delay tolerances of each of the traffic types that are being routed. Therefore, long offset times that violate delay tolerances of a given burst are not ideal. Offset times should therefore be long, but within limits of the delay tolerance of the burst.

- Destination

Each egress node has a minimum offset time associated with it from a given ingress node, which translates to a minimal delay tolerance of a burst. It should be that, there are no packets with delay tolerances less than the minimal offset time associated to their destination.

- Routing

The choice of a taken route must inevitably take into account the available offset and delay tolerance of a burst. Dynamic Routing (DR) is used for its flexibility. Other methods are proposed in [28], where rerouting is done from the source. There are many reasons why this method might not be desirable, one of them being that signalling for such a procedure would be complicated. Another is that this might require buffering all transmitted data before it is received.

When setting offset times, the possible routes must be taken into account. It is therefore desirable for a set offset time to accommodate possible rerouting. The assigned offset would therefore not be the minimum time for the shortest route or be too long that delay tolerance of bursts is violated. For rerouting, offset time should be preferably long.

Table 1 shows a summary of the considerations to be made during offset assignment.

Table 1. highlights the desirable offset times for each of the above factors and also suggests that offset time should managed in a flexible way.

OFFSET FACTOR	PREFERRED ASSIGNED OFFSET
High priority	Long
Delay tolerance	Variable
Destination	Fixed
Routing	Preferably Long

4.1.3 Offset assignment and burst assembly

From discussions in sections (4.1.1) and (4.1.2), we deduce that offset assignment and burst assembly schemes are complementary. A burst class is assembled with QoS requirements, and offset times are assigned to bursts to meet these requirements. In this section we discuss this relation.

- Destination D,

The shortest hop distance to a destination D determines a shortest offset time that can be set for a burst. For every assembled burst, the available delay tolerance, if partitioned appropriately between assembly time and transmission time, should always be enough to guarantee enough offset time for routing through the OSPF route. As such, $T_{OSPF} < T_{minD}$ is always true where T_{OSPF} is the time for taking the OSPF, and T_{minD} is the minimum delay that corresponds to the minimum burst length. Cases when this condition is not true are when upper layer protocols have too much latency.

- Offset time and delay tolerance

The assembly time is inversely proportional to the offset time, that is, the longer the assembly time, the less time will be available for setting an offset at the burst queue. It has implications on delay of traffic and therefore on QoS as well. While the burst is being assembled, the BHP is created. It is scheduled and transmitted when assembly ends, if no other bursts are in the burst queue and then begins to reserve resources for the burst.

Besides packet drops due to buffer overflow, QoS also depends on the delays incurred by the packets, from the moment of entry into the ingress to the time of exit at the egress node. Each traffic type has a delay tolerance range within which transmission of the packet should be done. Table 2 shows several applications that we consider in this study to represent Internet traffic.

Table 2. showing different traffic types and their delay requirements.

TRAFFIC TYPE	ONE WAY DELAY TOLERANCE (MS)
Video conferencing	less than 80
Conversation	0-≤80 (by ITU-T), 0-150 (Preferred), 150-400 (with degradation in Quality)
Video streaming (MPEG)	150-400ms

Audio streaming	10ms
Two-way control Telemetry	250ms
Interactive gaming	250ms
Web-browsing	2-4 secs per page, 0.5secs (preferred)
E-commerce	2-4 secs
SMS, Fax	30 secs
e-mail	Greater than 1min

In this study we use the different delay tolerances of the packets they have at the time of arrival at the ingress node to assign priority to the packets. To assign priority for each of the packets, we assume there are infinite priorities. We use equation (6) to assign each packet a priority $q_p(t)$.

$$q_p(t) = (t - \tau)b_p \quad (6)$$

Equation (6) is proposed for time shared systems in [50], where τ is the time of arrival of the packet, t is the current time of service, and b_p is a rate of service of the packets. In this study we let b_p be equal to the inverse of the assigned delay tolerance of the packet. We use this method of assigning priority at both the packet and burst buffer of the queue network.

4.2 The packet queue

We model the packet queue as GI/G/1/m queue i.e geometrically identical independent inter arrival times, with general service rates. GI is an adequate model of the self-similar inter-arrival periods of the packets to the LER. The service of the packet queue is synchronized against both the rate of arrival λ_T to the rate of assembly μ_A of the assembler. Where λ_T is the

sum of T independent Pareto ON/OFF streams, and μ_A is the overall of the service rate of the server. For optimal performance, $\lambda_T < \mu_A$ to avoid infinite growth of the packet queue, resulting in buffer overflow. μ_A is matched against the service rate of the packet queue. μ_A can be rated in terms of number of packets served per unit time with reference to the packet queue, or rated in terms of number of bursts per unit time with reference to the burst queue.

The lengths of the packet queue must therefore be continually monitored to ensure that the packet buffer does not overflow. In case of unavoidable packet queue overflow, there must be a packet drop policy. The packet drop policy must take into account the demand of QoS of each of the packets. To cater for QoS, packets are identified by their delay tolerances. We identify four classes of traffic in the Internet:

1. Traffic with very short delay tolerances, such as live video conferencing traffic. We label these as class 1.
2. Traffic with delay tolerances that do not have very strict QoS constraints, such as VOIP. We label these as class 2.
3. Traffic with long delay tolerances, such as SMS applications. We label these as class 3.
4. Background traffic such as e-mail. We label these as class 4.

4.2.1 Size of buffer and utilization

The size of the buffer must be large enough to accommodate the large bursty periods. The packet queues can be in four of the following states:

- Empty

A queue can be empty, when the packets in the buffer have all been burstified. All four queues may be empty if the LER is not receiving any packets. In this case there should be an error in the network. This scenario should be highly improbable during normal network operation.

- Normal

We describe the packet queue to be normal when the queue is not empty, and less than 80% of the maximum queue length. During the periods through which a normal packet

queue persists, the packet assembler as well as burst queue are said to be in equilibrium at the LER.

- **Almost full**

We describe an almost full state when the queue lengths are over 80% of the maximum lengths. At packet queue lengths greater than 80%, we call this an unsustainable equilibrium state. The existence of this state implies the continual growth of the queue over a period of time, without reaching steady state equilibrium. To prevent further unsustainable queue growth, a drop policy should be implemented. Several drop policies are possible, we use a fair queue drop policy.

- **Full**

We describe a queue to be full when the queue length is 100% of the maximum length. We call this an unstable queue state that leads to inevitably dropping of packets. To correct this state, a rapid drop strategy is implemented using fair queuing.

We now express the four states of the buffer with a mathematical and analytical understanding. Consider for starters a single ON/OFF Pareto source of traffic. We make the following critical assumption:

With the buffer being empty or nearly empty, a single ON period cannot cause it to overflow.

The implication with this assumption is that for all ON-periods, a prediction can be made to change the service rates so as to accommodate future packets into the buffer.

The first variable that needs to be established at the packet queue is the state of the buffer at the time of arrival of the packets. Let X_{ON} and X_{OFF} be a series of ON-OFF respectively periods of the single source. Then for the Pareto distribution, Let P_{ON} and P_{OFF} be the probability distributions of the ON-OFF periods respectively. Let the ON-periods be assigned the value of 1, since the rate of the ON-period is uniform, and 0 for the OFF periods. Then, at a given time instance of packet arrival, the rate of arrival at the buffer is given by the arrival rate 1, and during

the idle times, the arrival rate can be assumed to be 0. The current rate of arrival into the buffer would then be given by:

$$X(t) = \begin{cases} 1 & \text{if } \sum_{i=1}^{n-1} (X_{ON} + X_{OFF}) \leq t < \sum_{i=1}^{n-1} (X_{ON} + X_{OFF}) + X_n, n \geq 1 \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Which is the current state of the buffer before the X_n^{th} state arrives at the buffer. Because during a complete period of ON-OFF, packets are only received during the ON period, $X(t)$ then indicates that the source is ON. Several aggregations of $X(t)$ sources would then result in self-similar traffic. The state of the buffer will change depending on the service rate at time t . Let $r(t)$ be the service rate at time t . Then, if x is the state of the buffer at time t , then the service rate at state x will be $r(x)$. $r(x)$ is then given by:

$$r(x) = \begin{cases} r & \text{if } x > 0 \\ 0 & \text{if } x = 0 \end{cases} \quad (8)$$

Equation (8) shows that as long as there are packets in the buffer, then they will be served at a rate r , and if there is nothing in the buffer, the rate will be zero. After knowing the rate at which the packets arrive and are served, we need to determine an expression for the workload that exists in the buffer at a given time t . Let the load at a given time instance t be $L(t)$, given that $L(t)$ is less than the capacity of the buffer. Intuitively, the load $L(t)$ will be given by number of packets that remain in the buffer after a given number of packets are served in the same time interval. Therefore, the load at a given time will be given by:

$$\frac{d}{dt} L(t) = X(t) - r(L(t)) \quad (9)$$

Which gives: $dL(t) = dt(X(t)) - r(dtL(t)) \quad (10)$

With time, the load at the packet queue is expected to vary, depending on the ON and OFF intervals. During the OFF period, the load at the buffer can only decrease, and we can refer to these as inter-arrival times. During the ON period, depending on the size of period, and the rate of service, the buffer load may either increase or remain within the same limits. Consider an ON period with a series of bursty periods B_i , with $i > 0$, the load will increase by B_i during the

i^{th} interval. The effective increase in load during the i^{th} interval, $L_{inc,i}$ at the buffer would be given by:

$$L_{inc,i} = B_i + X_i - r_i X_i \quad (11)$$

Knowing $L_{inc,i}$, then allows for the estimation of the service time interval for a burst B_i at a rate r_i . This will be given by;

$$s_i = \frac{B_i}{r_i} \quad \text{for } r_i \geq 1 \quad (12)$$

Thus, the waiting time of a given burst B_i will be the sum of the service times of the traffic bursts in front in the queue. From (12), it is possible to introduce QoS. During the OFF periods, the buffer load will decrease by a rate relative to the rate of service. The rate at which the B_i bursts arrive at a node must be less than the service rate r_i for stability.

The distribution of the transmission times are dependant on the distribution of the burst lengths and offset times, if the offset times are random. Offset lengths are dependant on the destination of a burst and its delay tolerance and independent of the burst lengths.

4.3 Field Equivalence Classification of Packets into Bursts

Each of the packets entering the packet buffer has QoS requirements that are independent of the other packets in the buffer. From the discussion of the considerations made in the proposed scheme, we discuss how the packets are classified into the four FEC burst classifications proposed in this study. We use decision theory [51] to classify the packets into the appropriate burst. The decision making is based on the fact that every packet has a delay tolerance that allows for flexibility during packet routing, and on the assumption that the no packet has a delay tolerance less that the amount of time it takes to route the packet through the OBS network.

We choose four FEC classifications to match the 4 different types of packets. A class 1 packet does not necessarily mean the packet will be assembled into an FEC burst with the highest priority. We shall refer to the four different packet types as packet type 1, packet type 2, packet type 3, and packet type 4. We shall also refer to the four different FEC classifications as FEC A, FEC B, FEC C and FEC D with FEC A giving priority to the shortest delay tolerant

packets in the packet buffer, and FEC D giving priority to the longest delay tolerant traffic. The decision as to which FEC a burst should take is made as the burst is being assembled, and a final decision made when the burst has been fully assembled.

We shall use the symbol \succ to indicate strict preference of one FEC to another and \succsim to indicate relative preference. APPENDIX F. On arrival into the packet buffer, packets have their individual FEC preference, and each packet's preference is first considered separately.

Once assembled, the packets, though initially with different FEC preferences, take on a single FEC preference which falls within range of their delay tolerance. The FEC classification we make takes on two stages. First the packet's FEC preference is determined from its delay tolerance, depending on the joint FEC preference of the partial burst being assembled. The packet may then either be given its preferred FEC choice or a different FEC that does not violate its QoS requirements. On the other hand, should the packet be the first burst being assembled, then no comparisons are made. Packets' FEC preferences can also be easily determined through the Type Of Service (TOS) specifications and destination fields in the packet header. We give packets the following strict FEC preferences:

Table 3. Packet types and their correspondence FEC preferences

	FEC Preference
Packet type 1	$A \succ_1 B \succ_1 C \succ_1 D$
Packet type 2	$B \succ_2 A \succ_2 C \succ_2 D$
Packet type 3	$C \succ_3 B \succ_3 A \succ_3 D$
Packet type 4	$D \succ_4 C \succ_4 B \succ_4 A$

Table 3 strict preferences show Packet type 1, first prefer to be classified in FEC A then FEC B if the former is not available, then FEC C and finally FEC D. An FEC may not available if a packet requesting for the FEC finds packets of the same or higher priority in the queue ahead of it that add up to more than the maximum burst length of the current burst being assembled and the waiting time in the queue is longer than the delay tolerance of the packet. In case of delay tolerance violation, the packet is dropped. Packet type 2 preferences are first FEC B, which adequately serves their QoS demands, with a secondary option of FEC A, which would still

maintain the QoS demands, then FEC C and FEC D respectively are the next preferred FECs, but may not guarantee QoS. For Packet types 1 and 2, the preferences of FECs C and D are on condition that the burst being assembled will be transmitted in a time limit within the Packet 1 and 2's requirements. This will involve pre-empting the transmission of a burst at the LER during assembly. The set offset would then have to be set to meet the requirements of the shorter delay tolerant packets. FEC preferences of Packet type 3, are FEC C, which adequately meets its QoS requirements, then FEC B and A, which still meet its QoS demands, and finally FEC D if available, which would have to be pre-empted for transmission. Packet types 4 have preferences in the reverse order of delay tolerance, and dependent on the availability of the FECs.

Strict FEC preferences do not allow for flexibility during assembly. Consider a case where at time t , a burst of FEC C assembled to a length x , greater than the minimum required burst length but less than the maximum burst length, with timeout for burst assembly not having expired. The assembly scheme should allow for pre-emption of burst transmission, or padding of the burst with packets of a different packet type. In this study we assume that the packet queue is never empty, therefore the latter condition is applied. Padding of bursts allows for flexibility of the assembly routine. To accommodate burst padding, we have to use relative preference as well. We therefore expect the average burst lengths of the bursts to be longer as a result of padding. Table 4 shows packet type preferences.

Table 4. Packet type preferences giving relative preferences

	FEC Preference
Packet type 1	$A \succ_1 B \succ_1 C \succ_1 D$
Packet type 2	$B \succ_{\sim 2} A \succ_2 C \succ_2 D$
Packet type 3	$C \succ_{\sim 3} B \succ_{\sim 3} A \succ_3 D$
Packet type 4	$D \succ_{\sim 4} C \succ_{\sim 4} B \succ_{\sim 4} A$

Table 4 shows that only packets with high delay tolerances will take preferences to higher order FEC classes which would still not violate its QoS requirements. For instance, Packet type 3 will be allowed to pad FEC B and A if and only if they are idle and timeout for burst transmission is pending. Should padding options not be available, packet type 3 then takes the option of FEC D, which may result in the burst being pre-empted for transmission. In contrast,

packet type 1 is not allowed to pad lower priority bursts, as that may result in pre-emption of very short bursts in the core node, which would increase burst drop probability.

The proposed FEC scheme results in the assembly of bursts that may have multiple packet types while guaranteeing the QoS for each of the individual packets. By meeting the QoS demands for each of the packets, we do not compromise network performance, therefore, all bursts transmitted must have the minimum and maximum burst lengths.

The flow chart in Fig 4.3 shows FEC classification of packets at the packet buffer in the LER. In the flow chart, x is an integer from 1 to 4.

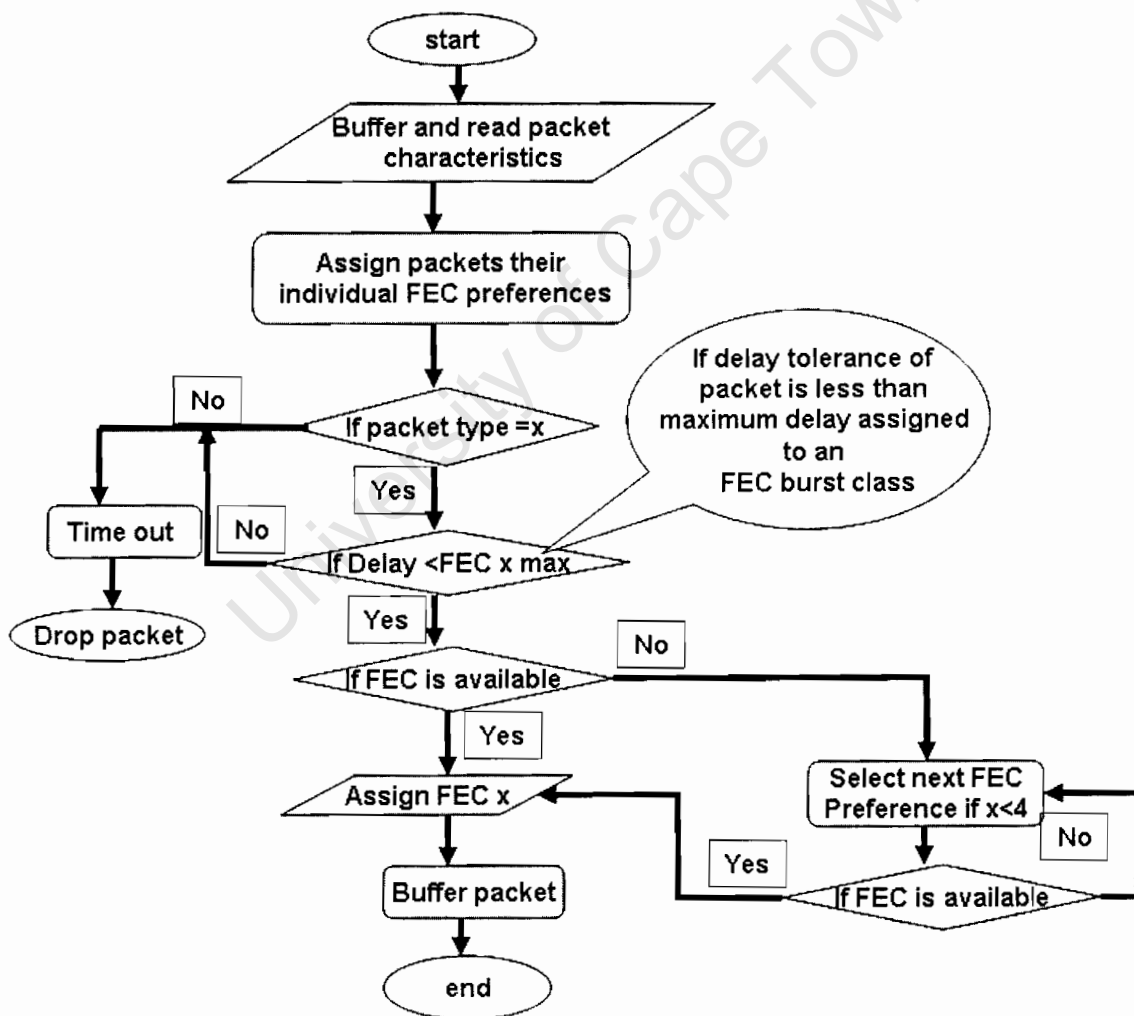


Fig. 4.3 Packet FEC classification scheme flow chart at the LER packet buffer.

4.4 The Assembler

We present the design of the performance of the assembler. We design an assembler to play the pivotal role of maintaining equilibrium between the packet queue and the burst queue. The assembler has a constant update of the number of packets that are in the buffer in the packet queue. The packet queue constantly informs the assembler of the number of packets queued, with respect to their classes, destination and delay tolerances. The assembler therefore admits packets from the queues with consideration of the required QoS for each class.

The four class queues at the packet queue end are served by the assembler. Each queue is served at a different rate, and independently from each other. The lengths of the bursts at the assembler are determined by the state of the packet queue.

The assembler must also take into account the capacity of the burst queue. In order for equilibrium to be maintained between the packet queue and burst queue, the assembler must have precedence over the burst queue. Take an instance when the burst queue is full with all four classes of traffic, and the packet queue is overflowing with high priority traffic. The assembler must be able to assemble the high priority packets into a burst and pre-empt to drop a class four burst in the burst queue. This would generally lead to higher overall QoS of high priority traffic.

Consider packets being served at a rate r by the assembler. For a burst of length l , the rate of burst generation is r_B , and is given by: r/l . The average rate of burstification should therefore be equal to or greater than the rate of arrival of packets into the packet queue. In turn, formed bursts must not be buffered before being moved into the burst queue. This is to limit redundancy in the queue, which would in turn increase delay in the LER. The rate of burstification should therefore correspond to the rate at which the bursts are being transmitted from the burst queue in order to maintain a stable burst buffer.

If we consider burstification in terms of time, then let a random variable $A(t)$ be the number of packets arriving at the assembler in a specified unit time. Then for a burstification interval of Δt the number of packets being assembled or in a burst would be;

$$A(t + \Delta t) - A(t) \tag{17}$$

If Δt is the time duration of assembly, then (17) would be the number of packets in a burst. If Δt is less than burstification period, then (17) is the number of packets in a burst still being assembled. Δt is always less or equal to the assembly time for maximum burst length and always greater than the time for burstification of a minimum burst length. . From (17), we can derive the rate of burstification r_B for a given time interval Δt , $R_B(t)$. $R_B(t)$ is a random burstification variable, which is;

$$R_B(t) = \frac{A(t + \Delta t) - A(t)}{\Delta t} \quad (18)$$

Fig 4.4 shows how the assembler handles the FEC classification of the assembled bursts. We do not however detail the performance of the assembly algorithm in this diagram indicated in Fig 4.4 as Assemble burst. A flow chart of the proposed assembly scheme is in Fig 4.5.

University of Cape Town

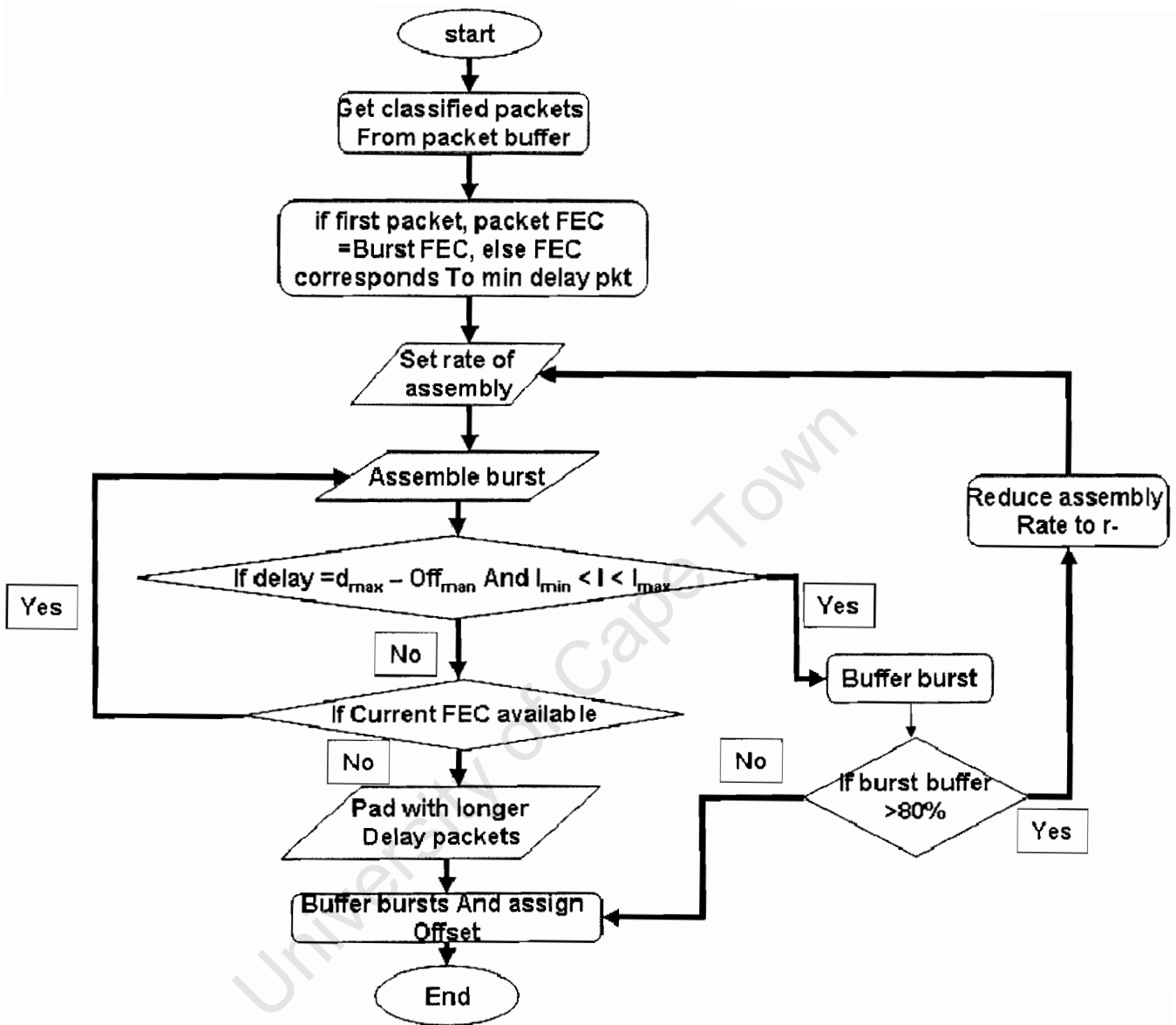


Fig. 4.4 Flow diagram of the proposed FEC classification, during burst assembly.

We propose an assembly scheme that combines the mixed-timer-length based scheme with the constant length scheme. Both schemes have their individual advantages, which are applicable to different packet classifications. We propose that FEC 1-3 packets be assembled with the mixed-timer-length based scheme due to their delay constraints, while FEC 4 packets, be assembled with the constant length based scheme since they have much longer delay constraints. An average of longer bursts in the OBS network will result in more efficient use of wavelengths, and reduce the overall number of bursts in the core. An overall reduction in

contention in the core is therefore expected. This method of assembly directly impacts on the delay of packets in the packet queue and burst drop in the OBS network further confirming the importance of having an optimal assembly scheme.

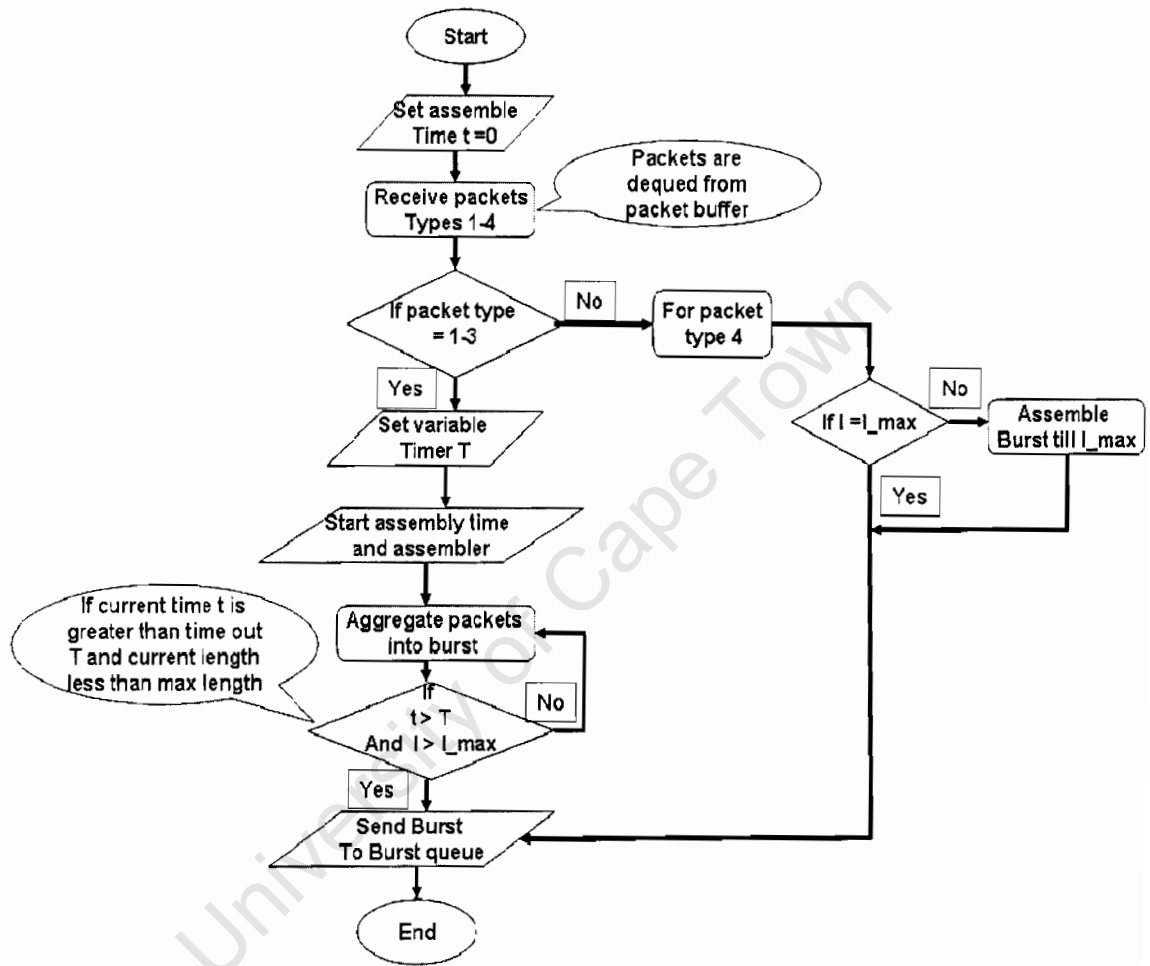


Fig. 4.5 Assembly procedure of the proposed FEC-assembly scheme.

We set the maximum burst length l_{max} , to 4000 kilobytes. However, instead of letting FEC 4 bursts be of strictly constant length, we give an allowance of 1000 kilobytes for packet types 1 to 3, *if available* to pad the FEC 4 burst. The maximum burst length of an FEC 4 burst will therefore be 5000 kilobytes with the last 1000 kilobytes being reserved. We allow only 1000 kilobytes in length addition due to delay constraints of these packets. By allowing 1000 kilobytes extra, delays of packet types 1-3 are further reduced.

The addition of the extra 1000 kilobytes may however complicate routing in the OBS network, since the delay for an FEC 4 burst would then be dictated by the lowest delay of the extra 1000 kilobytes FEC 1-3 packets. For QoS routing of an FEC 4 packet greater than 4000 kilobytes can be taken into consideration to avoid packet delay violation.

4.5 The burst queue

From the designs of the packet queue and assembler, we derive a design for the burst queue. The burst queue is serviced by the wavelengths available at the LER. We consider an LER with DWDM and wavelength conversion capabilities.

Service time S_t for the burst queue will include setting of an already assigned offset time to a burst and the transmission time. The offset time will depend on the FEC classification proposed, while the transmission time will depend on the duration of the burst. We avoid pre-emption during the transmission of both BHPs and bursts since OBS is a one way protocol.

- **Rule 1:** *After a BHP corresponding to a lower priority burst has been transmitted, the burst to be transmitted may not be pre-empted by a higher priority burst.*

There are four events that take place at the queue;

1. Assembled bursts with delay tolerance restrictions, each with a specific destination, arrive at the queue
2. An assigned offset is set for a given burst in the queue
3. BHP packets are transmitted to set an offset for a burst in the burst queue
4. The burst is then transmitted after the offset is set

The choice as to which burst is transmitted from the queue depends on delay tolerance of a given burst assembly, the arrival rate of classes, and the required offset times.

4.5.1 Delay tolerance

Rule 2: *If there is a contention for a wavelength between a low priority and high priority burst, a burst with a short delay tolerance will be served before a burst with a longer delay tolerance*

Unlike at the packet queue, we do not queue the bursts into four classes. Bursts are instead served depending on their delay tolerance which is derived from the FEC classification of the burst. A burst with long delay tolerance such as an e-mail burst will be transmitted after voice traffic burst. This rule also applies for cases when the longer delay tolerance burst has been waiting for service before the shorter delay tolerance burst. Rule two avoids a strict FCFS queuing routine and maintains QoS of bursts.

4.5.2 Class arrival rates

Rule 3: *The rate of arrival for any class into the burst buffer must not exceed the rate of transmission.*

At given instances, one traffic type may be more prevalent compared other traffic types. If delay tolerance is the only factor used to transmit, after a given period of time, one burst type destined for a given destination or destinations, may lead to an overflow of the buffer. To avoid an overflow due to one traffic type, the rate of transmission of the traffic type must be increased by a given factor to ensure that the throughput of the class is less than one.

4.5.3 Required offset

Rule 4: *More than one offset may be set for a given time slot to allow for void-filling.*

Void-filling allows for efficient use of the available wavelengths, and minimizes delays of the bursts in the queue. Consider three bursts in the queue. Let the assigned offset times be t_1 , t_2 and t_3 , for bursts with burst lengths b_1 , b_2 and b_3 respectively.

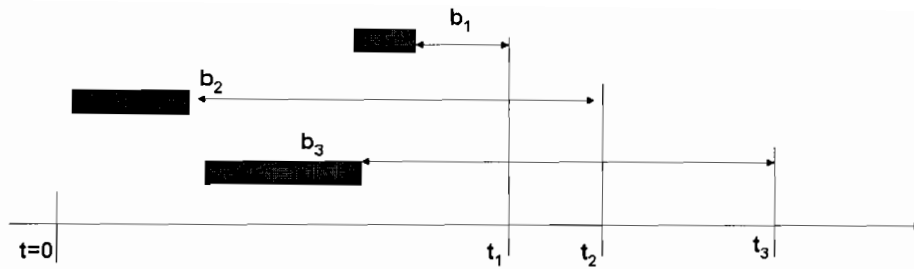


Fig. 4.6 An LER scheduling scheme that allows for void filling at the LER.

From Fig 4.6 it can be seen that burst length b_1 can be transmitted during the setting of the offset time of burst length b_3 . While the offset time of burst length b_2 can be set simultaneously with that of burst length b_3 , but only if the burst can be transmitted after length b_3 .

4.6 Assembly Queue Network

In this section, we discuss all three sections of the assembler; the packet queue, the assembler and the burst assembler. We derive the assembly network from Jackson network theory. In the network queue, we do not take into account arrival or departure of packets or bursts in between the two queues.

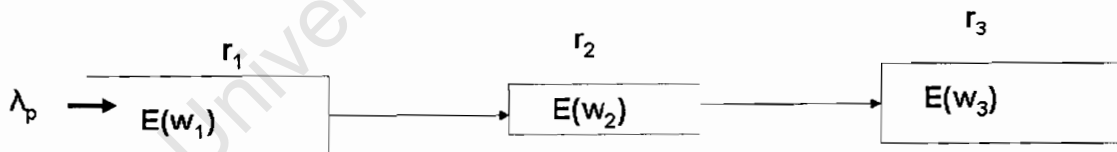


Fig. 4.7 A Jackson network from which we derive the formulation of the queuing at the LER.

From Fig 4.7 we let the arrival rate at the packet buffer to be λ_{mp} with m representing the number of aggregated p Pareto distributed traffic streams. We let the packet buffer be served at the rate μ_a by the assembler. Assembled bursts are transferred to the burst buffer at a rate λ_{bh} . The burst buffer is served at a rate μ_{wh} , where bursts are transported on different wavelengths. For the assembly network to be stable the ratio $\frac{\lambda_{mp}}{\mu_p} < 1$. However this must be done with the

burst queue stable as well, with $\frac{\lambda_{bh}}{\mu_{ub}} < 1$. The equilibrium of the queue network is controlled and determined by the rate of assembly, and the lengths of the offset times. Since the rate at which the packets arrive at the LER is an independent event from the rate of assembly, it is important to prevent the packet buffer from overflowing, without compromising both the assembler and the burst buffer. Consider a packet buffer that is 80% full, the probability that it over flows is high. To maintain an equilibrium, either the assembler, or the burst service rate μ_{ub} or both would have to be increased.

Increasing the rate of the assembler will reduce the probability of over flow of the packet buffer, while increasing the probability of overflow of the burst buffer. Increasing the rate of μ_{ub} , will decrease the probability of overflow of the burst buffer. The combination of both actions would therefore allow for controlling the equilibrium state of the queue network.

There are three waiting periods at the LER, at the packet queue, the assembler, and burst queue as shown in Fig 4.2. The duration of waiting times of the individual packet is mainly dependant on the service rate of the buffers. Total waiting time for a given packet at the edge node is therefore given by;

$$E(w) = E(w_1) + E(w_2) + E(w_3) \quad (13)$$

The expected waiting periods of the packets at the packet queue $E(w_1)$ is limited to the delay tolerance D_k of the packet. We let a given fraction of the delay tolerance of the packet to be used in the packet queue. $E(w_1)$ is therefore given by;

$$E(w_1) = aD_k \quad (14)$$

Where; D_k is the total available delay tolerance of the packet, and a is a fraction of the available delay tolerance. The value of a is chosen to be less than the total delay tolerance of the packet, but must be more than the minimum time it takes to route a burst from the ingress node to the egress node. By pegging the expected delay to a fraction of the delay tolerance of the packets, we ensure that packets with stringent QoS demands are identified and given higher priority.

The expected waiting period $E(w_2)$ at the assembler depends on the rate of assembly of the assembler, r , and the length of the burst being assembled. $E(w_2)$ is therefore given by;

$$E(w_2) = \frac{E(l)}{r} \quad (15)$$

Where $E(l)$ is the expected length of the burst, and r is rate at which the assembler assembles the packets into a burst. $E(l)$ is not an exact value due to different delay tolerances of the packets. The expected length of the bursts increases to a maximum value, above which it cannot exceed, so as to maintain efficiency during the switching in the core.

The expected waiting period $E(w_3)$ is limited by the remaining delay tolerance after the $E(w_2)$ and $E(w_1)$ delays. Depending on the remaining delay tolerance and destination, priority is assigned to the bursts. $E(w_3)$ is given by;

$$E(w_3) = \text{Re } m \quad (16)$$

Where $\text{Re } m$ is the remainder of the delay tolerance of any packet in the burst with the minimal delay. It should be that there are no circumstances under which $\text{Re } m$ is less than required offset.

4.7 Summary of proposal

Fig 4.8 shows the resulting LER. Compared with the general LER shown in Fig 2.2, Fig 4.8 has more capabilities.

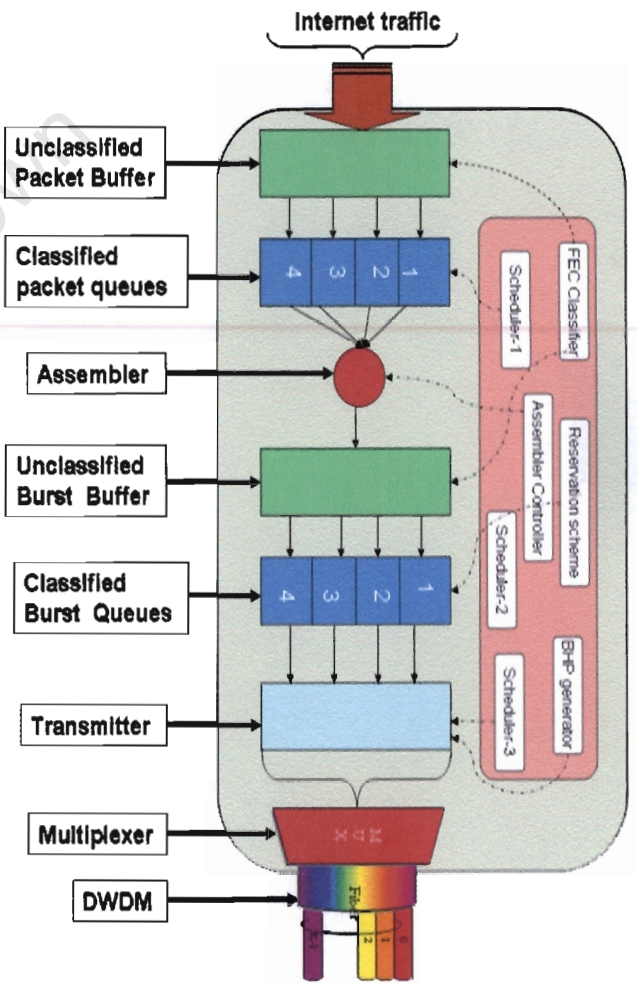


Fig. 4.8 The resulting LER from the proposals given in this work.

Chapter 5

5 Simulation Set-ups

In this chapter we describe the simulation set-ups used. For this work, the following environment settings apply:

Table 5. Simulation environment.

Computer processor	3 GHz Pentium 4 processor
Total hard disk space available	80GB
Operating system	Fedora Core 3 Linux operating system
Simulation platform	Ns-2.28
Programming languages	C++ and TCL
Base simulator	OBS.0.9a

The Ns-2.28 (commonly known as ns2) is a network simulator widely used for many network simulation scenarios. A lot of documentation exists, detailing its performance. One such documentation is in [51]. The OBS.0.9a [52] software is an OBS simulator, built to run on ns2. It is a very basic simulator, only giving functionalities of major OBS components, the LER, LSR, DWDM fibre, LAUC-VF and JET scheduling and reservation schemes.

To the OBS.0.9a simulator we added the following functionalities: Four assembly schemes, constant, timer, mixed timer-length, and the proposed FEC-Based assembly schemes,

two offset assignment algorithms, uniform and Pareto distributed offset assignment (see Appendix G for full description). We also add the two proposed queues, packet and burst buffers.

We set up our simulations to investigate performance delay and burst drop of the different burst assembly schemes on the proposed queue network. We demonstrate that performance of the assembly scheme is related to the assignment of offset times. Four assembly algorithms are simulated: burst length based scheme, time based scheme, mixed-time-burst length based scheme and a combination of fixed burst length and mixed burst length scheme, the proposed FEC-based assembly scheme.

Of interest is the performance of each of the assembly schemes under either exponential or self-similar traffic. Results in the difference in performance, be it improved or deteriorated in OBS would be of significant interest. If performance of the OBS network is deteriorated when Self-similar traffic is used, then these results would further assert the fact that self-similarity in Internet traffic is a critical factor in Internet traffic engineering. If however performance of the OBS network either remains the same or improves when self-similar traffic is used, then the assumption made in most literature that Poisson traffic can be used to assume self-similarity would be true.

In our proposal we proposed that the offset be set using a Pareto distribution. The reason for the choice of a Pareto distributed offset period over other distributions is based on the observation of how transmission of bursts is done. The choice of a Pareto distributed offset period also gives uniformity in analyzing Internet traffic. Traffic in the core can then be modelled with clear relation to traffic before assembly. We compare performance of the OBS network with all four assembly schemes using either constant offset periods, or Pareto distributed offset periods. To further give a better indication of the extent to which Pareto distributed offset periods suit OBS, we simulate a uniform distributed offset with all assembly schemes. We also simulate performance of the OBS network on different sizes of a network with four different topologies.

We expect to show and/or determine the following in our simulations:

1. If the distribution of traffic does indeed change the performance of the OBS network

2. Whether exponential traffic does underestimate burstiness of traffic resulting in better than expected results
3. Determine whether the proposed Pareto offset assignment scheme does indeed result in better OBS performance
4. Whether the proposed FEC-based assembly scheme that combines constant burst length and mixed timer length performs better than the three first proposed schemes
5. We should be able to show that the proposed assembly scheme results in better packet and burst buffer management
6. We should also show that events at the LER do affect the overall OBS network performance

5.1 Simulation Set-up 1

The simulation set-up in this section is used in determining the appropriate simulation parameters. These are simulation time and Pareto traffic settings of the Hurst parameter

With increase in simulation time, it would be expected for the number of bursts dropped to increase. Burst drops would also be dependent on the distribution of traffic being used. To determine an appropriate duration of simulation for the results and traffic distribution, we use the simulation set-up in Table 6. We also monitor the performance of the network, in case of Pareto distributed traffic, the effect of the shape parameter, and compare it to exponential traffic.

Table 6. Pareto and exponential traffic settings.

SIMULATION DURATION (SECONDS)	FOR PARETO TRAFFIC: SHAPE PARAMETER	EXPONENTIAL AND PARETO TRAFFIC SETTINGS
100	1.2 & 1.5-1.9	Rate: 1Gbps
500	1.2 & 1.5-1.9	Packet Sizes: 500-1000 (Random)

1000	1.2 & 1.5-1.9	Number of Connections: 35
1500	1.2 & 1.5-1.9	Topology Size: 8 LSRs and 3 LERs, with 3 data channels and 1 BHP channel per fibre
2000	1.2 & 1.5-1.9	ON/OFF periods: 40 ms

From results obtained in this set-up, we determine whether performance for short durations of simulations significantly changes with increase in the duration of the simulation. Should performance differ significantly as simulation duration increases, then for all future simulations, maximum possible simulation time will be used, subject to performance limitations of the simulation environment.

5.2 Simulation set-up 2

With traffic and simulation duration settings from simulation set-up 1, the aim of set-up 2, is to verify the assembly proposal and offset assignment made in this work.

Table 7. simulation scenarios simulated.

TRAFFIC TYPE	OFFSET ASSIGNMENT SCHEME	ASSEMBLY SCHEME
Pareto	Constant	Constant Burst Length
Exponential	Constant	Constant Burst Length
Pareto	Pareto	Constant Burst Length
Pareto	Uniform	Constant Burst Length

Exponential	Constant	Timer Based Assembly
Pareto	Constant	Timer Based Assembly
Pareto	Pareto	Timer Based Assembly
Pareto	Uniform	Timer Based Assembly
Exponential	Constant	Mixed Timer and Burst Length
Pareto	Constant	Mixed Timer and Burst Length
Pareto	Pareto	Mixed Timer and Burst Length
Pareto	Uniform	Mixed Timer and Burst Length
Exponential	Constant	Constant Length and Mixed Timer Burst Lengths
Pareto	Constant	Constant Length and Mixed Timer Burst Lengths
Pareto	Pareto	Constant Length and Mixed Timer Burst Lengths
Pareto	Uniform	Constant Length and Mixed Timer Burst Lengths

From results obtained with this set-up the combination that gives the better performance is determined.

5.3 Simulation Set-up 3

From set-up 2, we determine the assembly scheme that results in the least burst losses with increase in topology. We then investigate if the assembly scheme with traffic settings from set-up 1 with varying topology sizes. The aim is to determine whether the different offset assignment schemes still result in significant performance difference with increase in the OBS network. We use three topologies 1, 2 and 3 for this simulation.

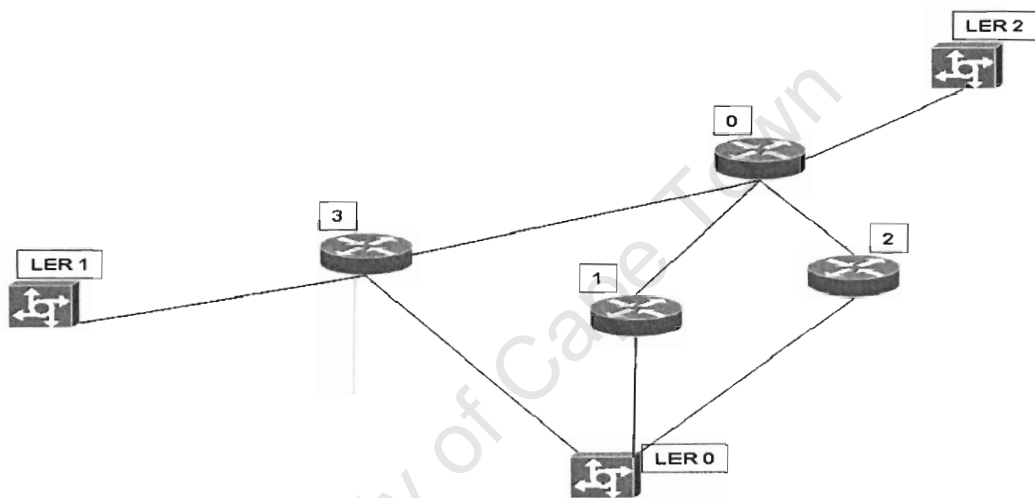


Fig. 5.1 A mesh Topology 1 with 4 LSRs, the smallest of the three topologies used for simulation.

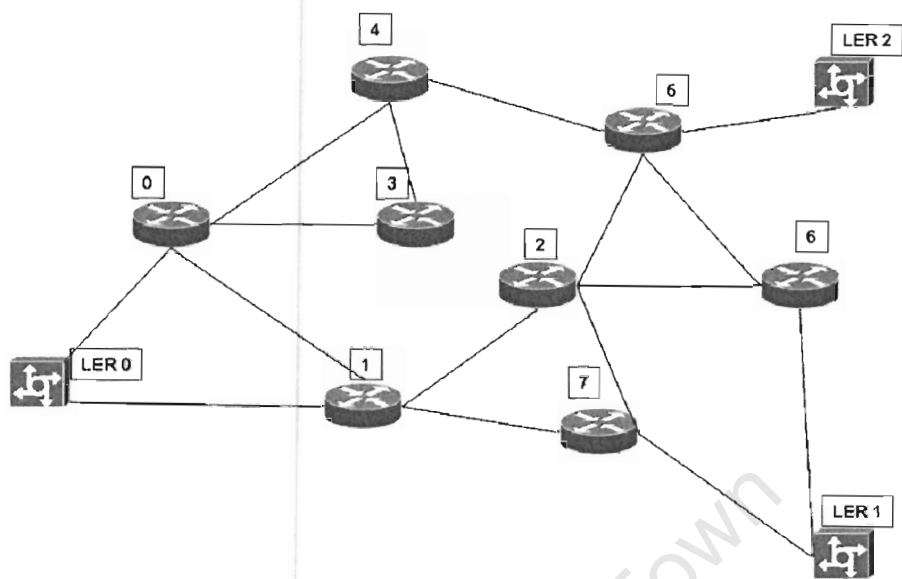


Fig. 5.2 A mesh Topology 2 with 8 LSRs used for simulation.

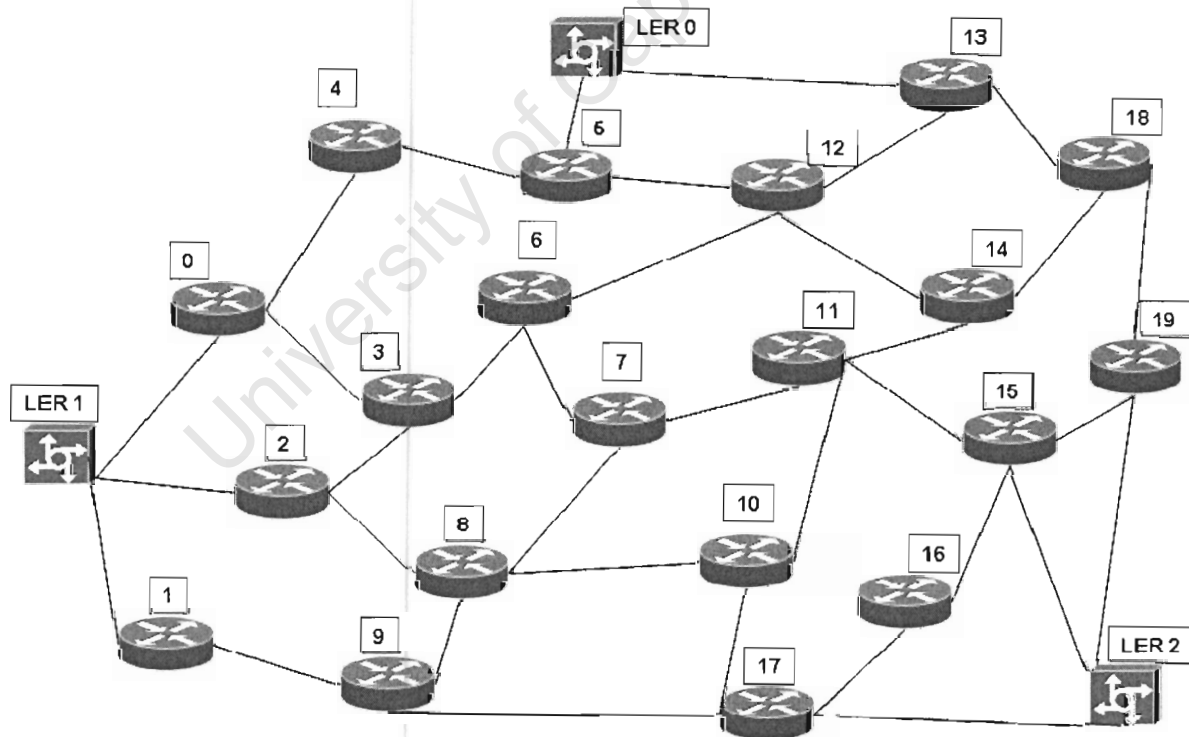


Fig. 5.3 A mesh Topology 3 of 20 LSRs, the largest of the topologies used for simulation.

Chapter 6

6 Simulation Results and Analysis

In this chapter, we present simulation results of the proposed assembly and offset assignment scheme and discuss their implications.

Previous assembly schemes and offset assignment schemes are contrasted with the proposed FEC-based and Pareto-offset assignment schemes. In the results we endeavour to show that traditional Poisson modelled traffic does not give a true indication of network performance or characteristics.

The simulation results presented in this chapter highlight the following aspects:

1. Traditionally exponential traffic underestimates burst drops in a typical OBS network
2. The proposed FEC-based assembly scheme out-performs the timer and mixed –timer-length based assembly schemes
3. The proposed Pareto offset assignment scheme results in the least burst drops compared to constant and uniform offset time assignment.

6.1 Results

The basis of this work is strongly hinged on evidence that self-similar traffic has a negative performance effect on a network.

To demonstrate behaviour of the simulator, we use traffic settings in Table 8 on a constant assembly scheme.

Table 8. Traffic settings used

Traffic setting	1	2
Packet sizes (bytes)	[500-900]	650
Average duration of bursty periods (seconds)	[0.01-0.05]	0.01
Average duration of idle periods (Seconds)	[0.01-0.05]	0.01
Rate (Kbps)	[20000-25000]	22000
Shape parameter (For Pareto traffic distributed traffic)	1.85	1.85

We use Table 9 to show that the different traffic settings result in different number of packets, and hence different bursts. The set-up used is for one wavelength per fibre on Topology 3. For each of the distributions we use random packet, rate and idle periods for a more realistic traffic behaviour. Simulations in Table 9 were done for only 135 seconds with a constant burst length assembly scheme.

Table 9. Showing the Pareto and exponential traffic settings send different number of packets

Traffic setting	Number of Packets sent	Number Bursts Sent	Burst Sizes (bytes)	Bursts Dropped	Connections per LER
1 (exponential)	1531	189	[3380-4000]	0	1
2 (exponential)	1080	180	3900	0	1
1 (Pareto traffic)	1563	189	[3380-4000]	0	1

2 (Pareto traffic)	1098	183	3900	0	1
--------------------	------	-----	------	---	---

To demonstrate predictable performance, we also simulate Table 9 settings, with random variable settings, for 270 seconds and 370 seconds. We expect an increase in packets and bursts sent.

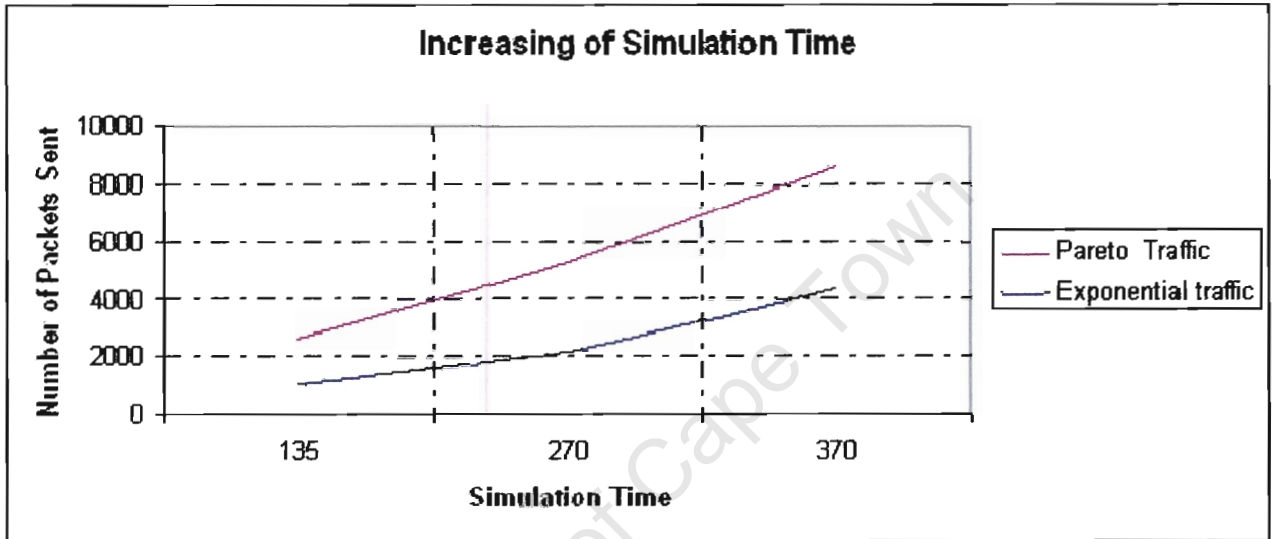


Fig. 6.1. Increasing of simulation time increasing number of packets and therefore the number of bursts

For simulation times 270 and 370 seconds, there were 363 and 520 bursts for exponential traffic and 380, 522 bursts sent for Pareto traffic.

To determine the performance (in terms of burst drops) of the proposed assembly scheme we have to compare its performance against the previous three schemes in the same situations. Measures of performance of the assembly scheme would be the number of bursts sent and the bursts dropped in the core network. We however have to first determine an appropriate duration of simulation that would give results that most accurately reflect network performance. We determine an appropriate duration for simulation and Hurst parameter from results in Fig 6.2 and 6.3 using set-up 1.

Fig 6.2 shows the increase in burst drop with increase in simulation time using the mixed-timer-length assembly scheme. It should be expected, as simulation time increases, so will the number of bursts dropped. The simulation serves to show that the simulation set-up of the OBS network is one that is realistic.

Fig 6.2 also shows that the increase in Hurst parameter of the incoming traffic into the packet buffer does not result in a drastic increase in burst drops. The reason for the small change in burst drop is because burst drops will be dictated more by the assembler, offset assignment and scheduling schemes. The Hurst parameter does however affect the rate at which bursts are assembled. The Hurst parameter for incoming traffic in the rest of this work is set to 1.85 [14]. The traffic settings were 35 Pareto traffic sources to per LER in topology 3, each at a rate between 2000KBps and 2500KBps.

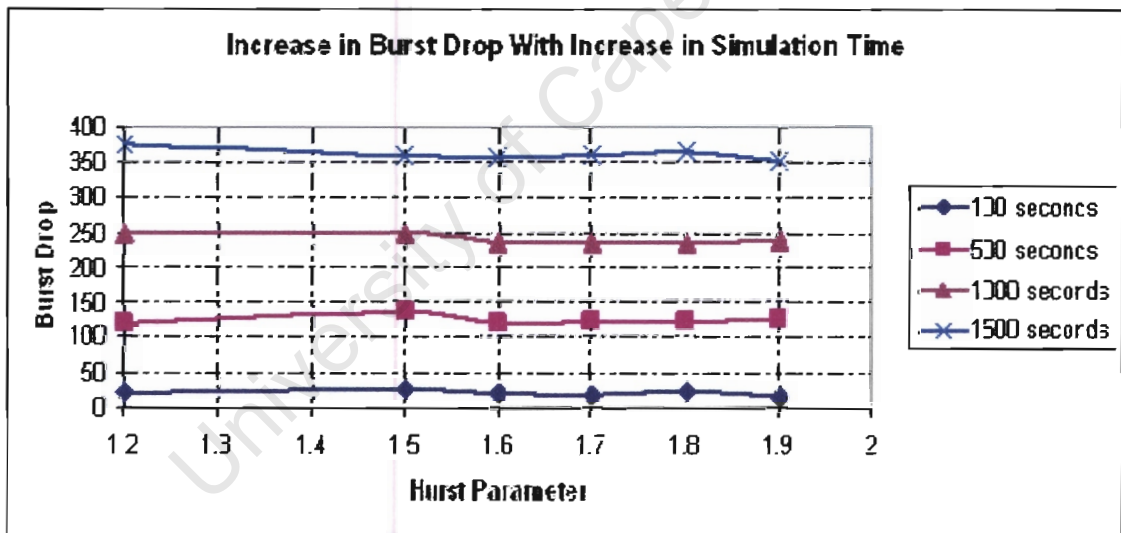


Fig. 6.2 The increase in burst drop with increase in simulation time.

A contrast of Fig 6.2, is Fig 6.3 showing the burst drop ratio against the Hurst parameter. Burst drop ratio is the number of bursts dropped, against the number of bursts sent.

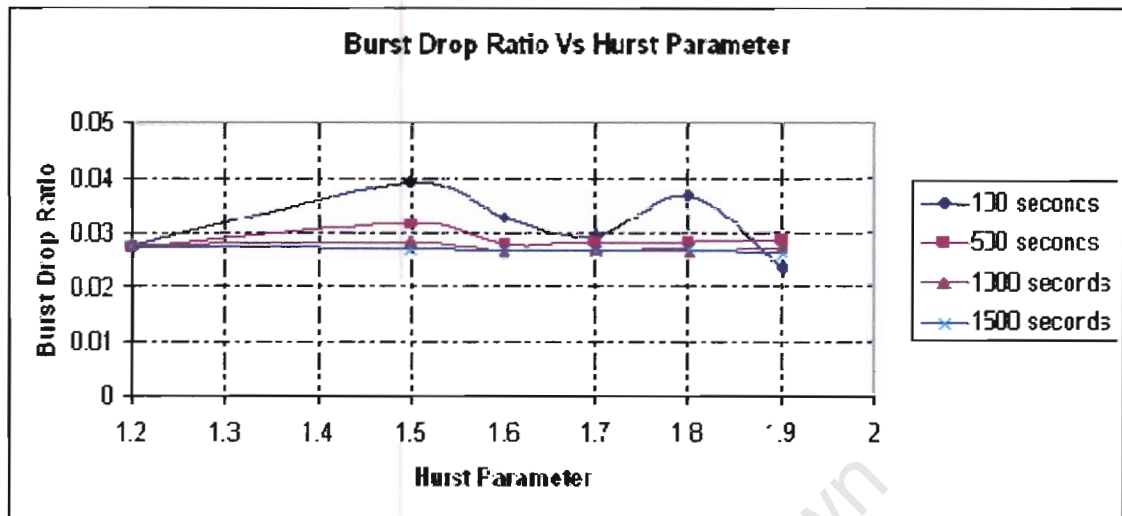


Fig. 6.3 The ratio of bursts dropped to bursts transmitted against increase in the Hurst parameter.

Fig 6.3 shows how the duration of the simulation may or not affect the ratio of bursts dropped in the OBS network. If the simulation duration chosen is short, in this case 100 seconds, the burst drop ratio does not give a true representation of the network performance. At 100 seconds duration, the burst drop ratio varies greatly, and simulation of the OBS network would give false results. However, as the simulation duration increases, the ratio of burst drops is about uniform. This simulation result is used to determine an appropriate duration for which simulation should be done. From Fig 6.3 a simulation time of between 1000 – 1500 seconds would yield more accurate results. For the rest of our simulation results we use 1500 for the duration of the simulation. For the durations of 2000 seconds and above, the simulation machine we use runs out of memory and stalls. Knowing the appropriate simulation parameters, we now test the performance of the proposed schemes.

6.2 Simulated assembly schemes

This section shows simulation of the timer, mixed timer-length and constant length assembly schemes. More traffic is used in this set-up, using 35 connections per LER on Topology 1. The 35 aggregated connections are for both exponential and Pareto traffic.

Fig. 6.4 shows assembly with a timer-based assembly scheme. For this simulation, we set the timeout to be 1ms.

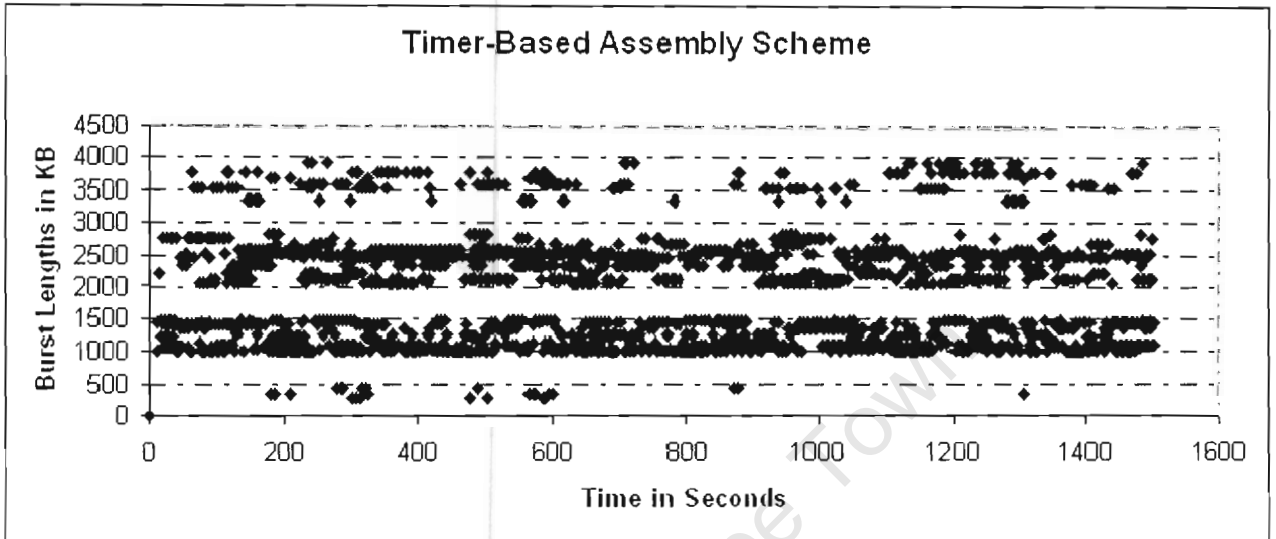


Fig. 6.4 Burst lengths when a timer based scheme is used for assembly without FEC.

Assembly with mixed timer-length based assembly in Fig. 6.5 has a lower limit of 3000 KB and an upper limit of 4000 KB.

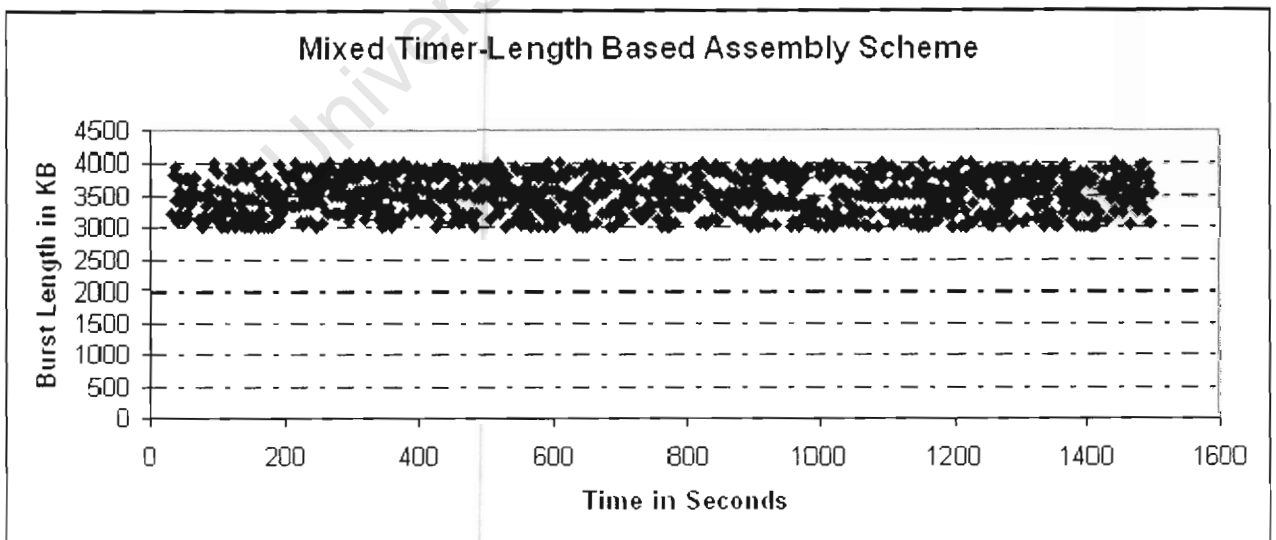


Fig. 6.5 Assembly with a mixed timer-length based assembly scheme

Fig 6.6 shows assembly of bursts on purely a constant burst length assembly scheme. Bursts are constantly at 3900 KB for constant packet sizes of 650 bytes. Each burst contains 6000 packets.

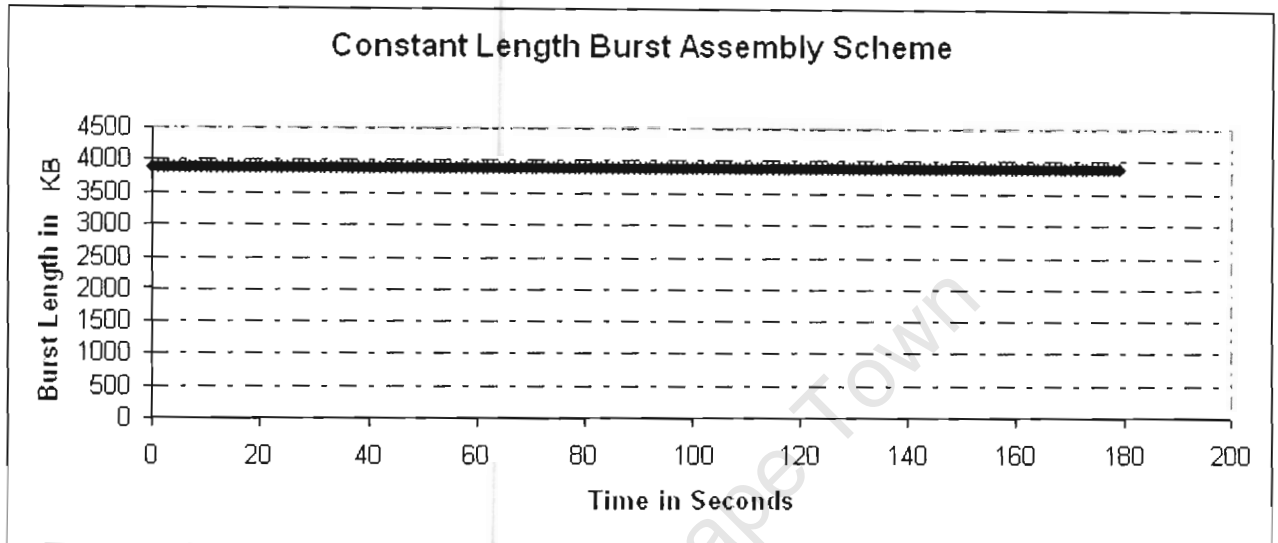


Fig. 6.6. Assembly of bursts with constant length assembly scheme with 650 packet sizes

We use the constant and mixed timer-length based assembly schemes for our proposed FEC-based scheme.

6.3 The Proposed FEC-based assembly scheme

The proposed FEC-Based assembly scheme results in bursts with burst length distributions shown in Fig 6.7. In Fig 6.7 we show a sample of bursts of FEC 4, and FEC 1-3. We allow for bursts of FEC 4 to have burst lengths up to 5000 KB, and a minimum 4000 KB. This is because for FEC-4 bursts, delay is not a major constraint, and burst lengths of up to 5000 KB can be assembled without violating time delay constraints. It should be noted that though FEC-4 bursts are assembled using a constant length based scheme, and burst lengths vary. The varying of burst lengths is due to the design of the assembly scheme, which dictates that when the threshold of 4000 KB of FEC-4 packets are assembled, and there exist in the buffer FEC 1-3 packets, these packets are used to pad the burst up to a maximum of 5000 KB in length. Pre-

emption must therefore occur if padding of packets with FEC 1-3 is done to avoid delay tolerance violation.

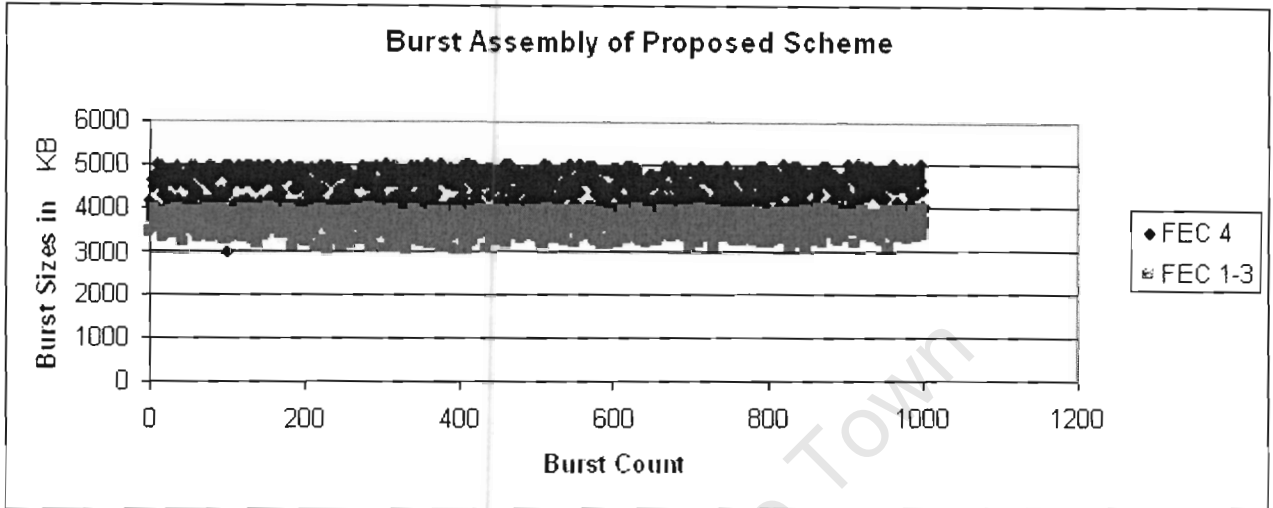


Fig. 6.7. Distribution of assembled bursts using the proposed FEC-Based Assembly scheme

The proposed FEC-based assembly scheme shows that less bursts are dropped compared to length, timer, and mixed-timer length based schemes. This result is expected since bursts in the FEC-based scheme are on average longer due to the isolation of FEC 4. Compare Fig 6.7 with Fig 6.4, where bursts are assembled using the timer based assembly scheme. Bursts from the FEC-Based scheme are generally longer than those from the timer based scheme. No FEC is used in Fig 6.4 and a minimum burst length limit is not specified. Generally, the longer the bursts, the less probability number of bursts in the core, more efficient use of wavelengths and therefore less resource contention.

6.4 Arrival distribution and contention at LSR

Further understanding of the effect of random offset assignment on contention in the core network is illustrated in Figs 6.8 and 6.9. Fig 6.8 shows arrival of bursts at an LSR node with a constant offset assignment and Fig 6.9 shows burst arrival at the same LSR, with a Pareto offset assignment. In Fig 6.8, bursts arrive at the LSR start arriving after about **10ms** of simulation. Bursts in Fig 6.8 arrive at very close time intervals about **10ms**. The close inter-arrivals result in

almost simultaneous burst arrivals at the LSR, resulting in burst drops due to lack of resources. In contrast the simulation in Fig 6.9 results in significantly less burst drops in the core nodes. The difference in burst arrival distributions at the LSRs can be observed. Bursts arrive in a more random manner with the Pareto-offset assignment scheme. We therefore conclude that the more random an offset assignment is, the less probable it is for a burst to contend for resources at an LSR. In turn, we affirm our argument that burst assembly and offset assignment have a direct impact on the performance of the OBS network. Contention is obviously not eliminated by the randomness of an assigned offset, rather contention is reduced. Schemes for resource contention may then be used to resolve remaining contention.

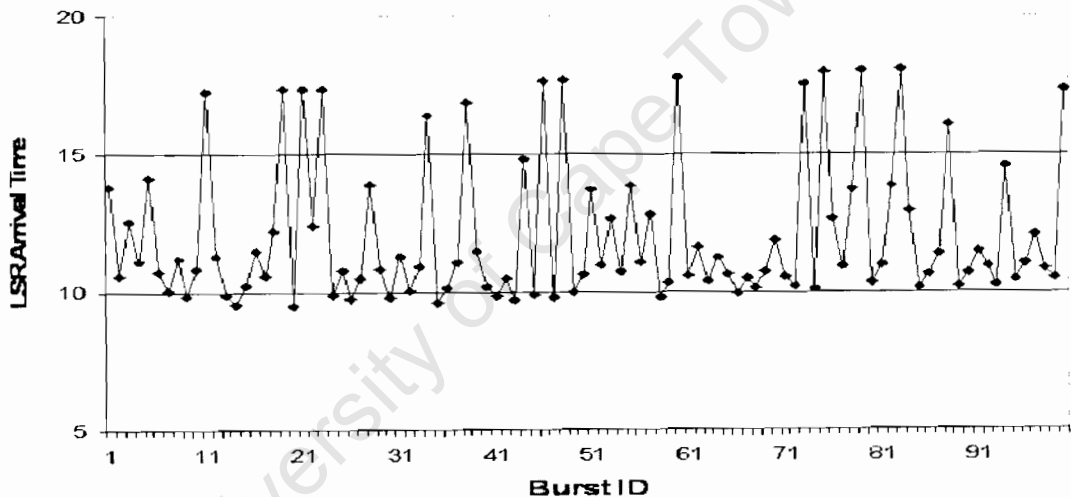


Fig. 6.8 Arrival distribution of bursts at an LSR 1, when a constant offset assignment scheme is used at the LER.

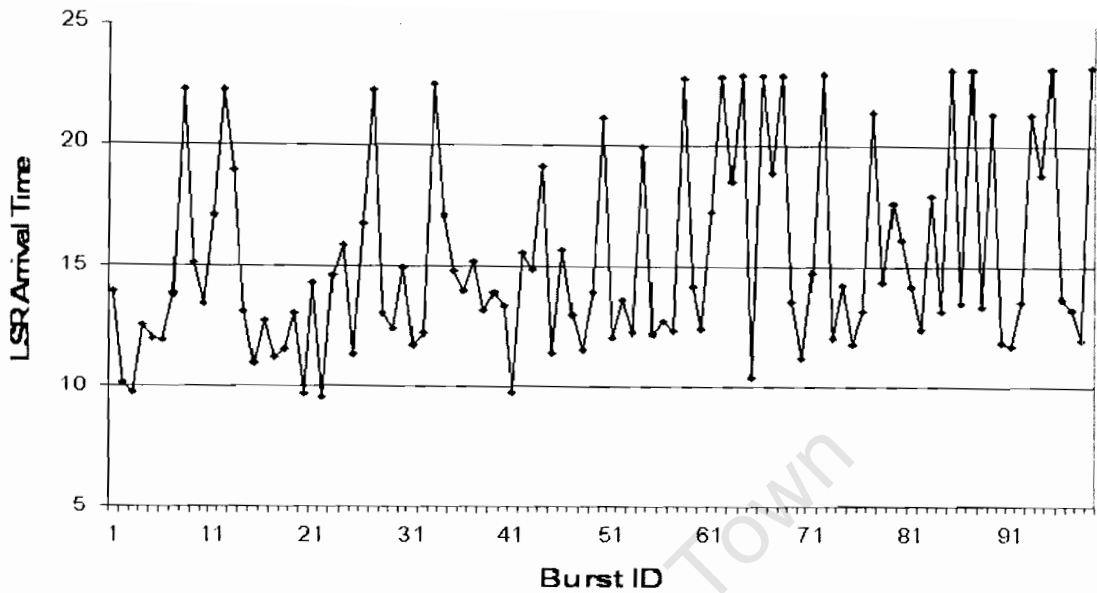


Fig. 6.9 Arrival distribution of bursts at an LSR, when a Pareto distributed offset assignment scheme is used at the LER.

6.5 Offset Assignment

Figs 6.10 and 6.11 show burst drops of the simulations with different offset assignment schemes. **Const-Pareto** represents constant offset assignment for self-similar traffic and **const-exp** represents constant offset assignment and exponential traffic respectively. The Pareto and Uniform bars in Fig 6.10 show Pareto and uniform offset assignment for self-similar traffic.

The burst drop percentage for exponential traffic in Fig 6.10 clearly underestimates burst drops for real Internet traffic depicted here by aggregated Pareto distributions to give self-similar traffic.

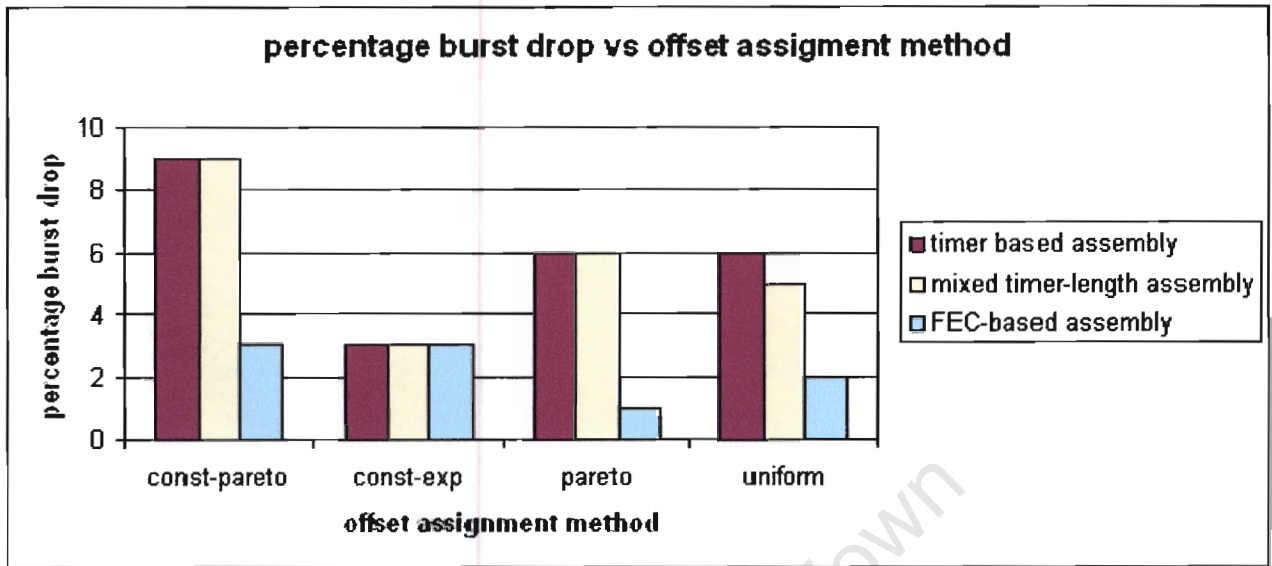


Fig. 6.10 Summary of the performance of the proposed assembly and offset assignment schemes using the proposed queue management method.

Fig 6.10 results show that assembly with the proposed FEC-based assembly scheme, there are less percentages of burst drops compared to the timer and mixed-timer based assembly schemes. For traffic settings in the simulations in this study, the timer-based and mixed timer based assembly schemes generally have the same performance. An important observation is that the combination of the proposed FEC-assembly scheme and the Pareto offset assignment result in best improvement by having the lowest number of burst drops.

In order to determine the effect on burst drop with the FEC-based assembly scheme, with increase in size of the topology, we use set-up 3. We contrast each of the offset assignment schemes to compare performance.

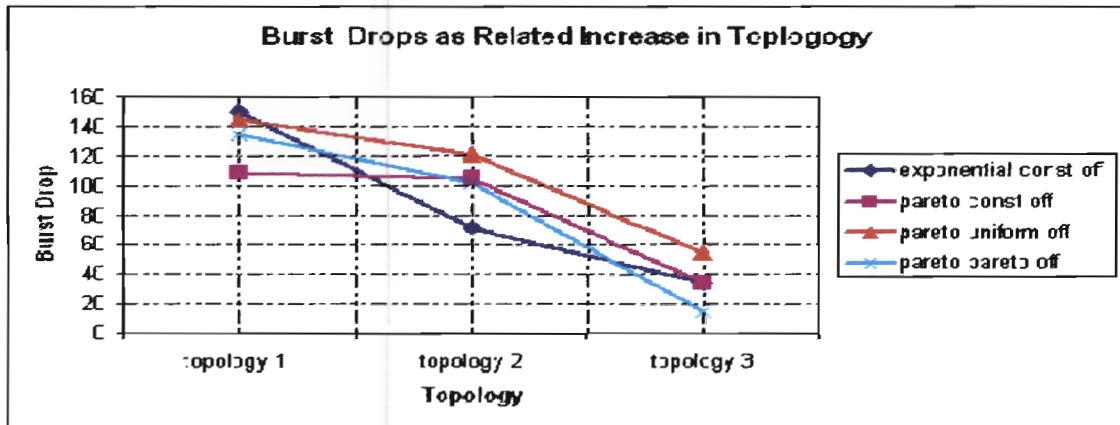


Fig. 6.11 Performance of the proposed FEC-Assembly scheme on both the Pareto and exponential traffic distributions and different offset assignment methods, with increase in the size topology.

Fig 6.11 shows that with increase in the size of the topology, there are less burst drops, as should be expected. This is because with increase in topology, there is an increase of number of routes available for bursts, and therefore more resources. The result serves to further show that the simulation settings are realistic.

6.6 Burst Buffer Delays

It can be intuitively deduced that, different methods of offset assignment have different a bearing on delay at the ingress node. The different delays would depend on the distribution and lengths of the offset times. In Fig 6.12 and Fig 6.13, we show the effect of different offset assignment on two different burst assembly schemes. Traffic settings for this section are detailed in set-up 1. The aggregation of 35 connections for Pareto distributed traffic result in self-similar traffic. Fig 6.12 shows delays incurred by bursts at the burst buffer using the timer-based assembly scheme. Fig 6.13 shows delays incurred by bursts at the burst buffer using the proposed FEC-based assembly scheme.

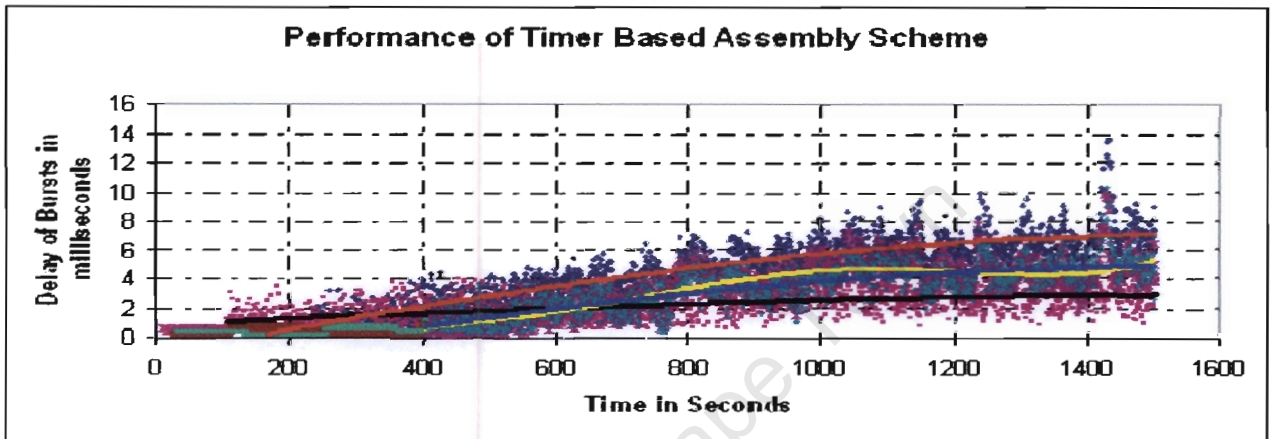
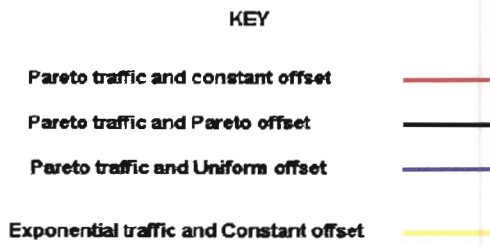


Fig. 6.12 How delay of assembled bursts varies at the LER when different offset assignment schemes are used on the timer-based assembly scheme.

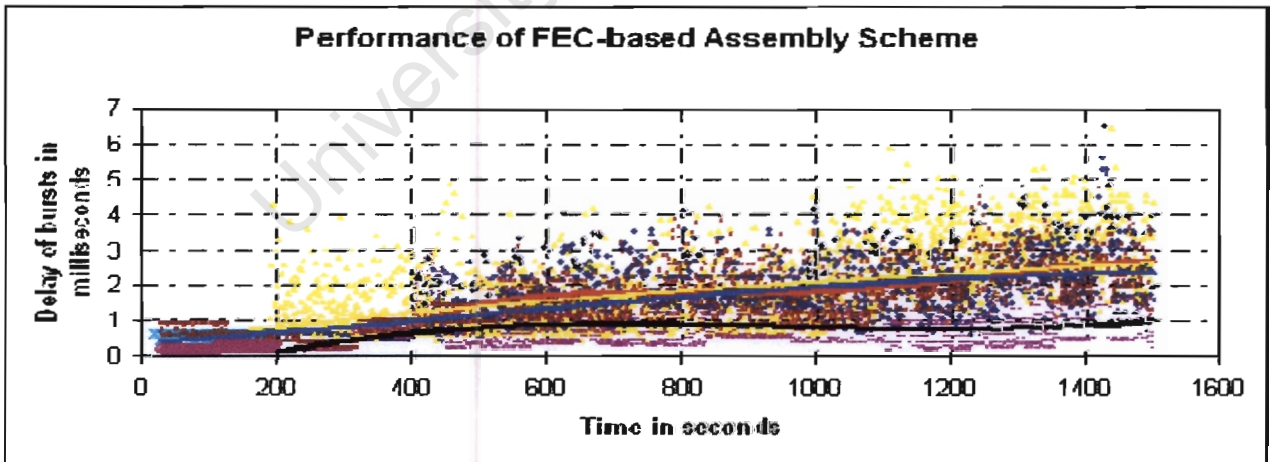


Fig. 6.13 How delay of assembled bursts varies at the LER when different offset assignment schemes are used on the proposed FEC-Assembly scheme

Fig 6.12 shows that for self-similar and constant offset assignment, bursts will incur more delay compared to exponential traffic with constant offset assignment. The reason for this is because exponential traffic underestimates the burstiness of the traffic. This result agrees with previous work in literature and from Fig 6.10. When offset assignment is made random rather than constant, for self-similar traffic, delay of bursts in the burst buffer decreases. The decrease in delay is because **Rule 4** can be implemented when random offset assignment is used, which results in more bursts being sent per unit time. From Fig 6.12, delays by uniform offset assignment are lower than delays for the same self-similar traffic with constant offset assignment. However, delays are further reduced by using the proposed Pareto offset assignment scheme. We account the different in delay to the more random variance of the Pareto distribution, compared to the uniform distribution.

In Fig 6.13, we instead use the proposed FEC-based assembly scheme. Overall delay of bursts is lower than that of the timer based assembly scheme. However, delay by constant offset and uniform offset assignments are not as different in this case, as is for the timer based assembly scheme. Notably though, is the lower delay of bursts when offset assignment is done using the Pareto offset assignment.

Chapter 7

7 Future Work

In this chapter, for future work, we propose, without results, routing and scheduling schemes for the already assembled traffic for future work.

7.1 Core Aggregation Scheduling Algorithm

OBS is a one way protocol. Efficient scheduling of the assembled bursts is needed, to avoid burst drop and bandwidth contention in the core network. The LAUC-VF algorithm, mentioned before, and others proposed after it, still encounter core resource contention. We propose a Core Aggregation algorithm with Void Filling (CA-VF) that promises better performance.

The CA-VF algorithm uses the same scheduling as LAUC-VF, but adds an aggregation feature. By aggregating payload bursts, core resources are more efficiently utilized.

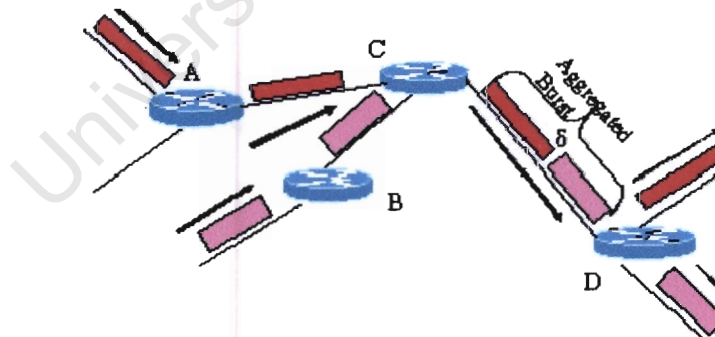


Fig. 7.1 Payload bursts arriving at LSRs A and B are routed to C. At C, the Bursts are aggregated depending on their time difference, δ , destination, $d1$, or $d2$, and/or the resources available at node C. The bursts may separate at D depending on their destinations.

Consider payload bursts at A and B, above, of different lengths. BHPs of each payload burst reach C and give information of their respective bursts. Therefore, before the payload bursts arrive at C, their lengths, l_{pb} destinations, d_1 , d_2 , and time differences δ are known. By knowing l_{pb} , d_1 , d_2 , and δ , payload bursts may be aggregated when, the conditions below hold:

1. When the routes being taken by the two bursts are the same for one or more nodes.
2. When δ is very short, such that, no significant time lapses while the wavelength is held for the next burst/s to be aggregated.

At node D, payload bursts may separate depending on their final destinations, in which case, two BHPs will be generated at D for each of the payload bursts.

By aggregating payload bursts, the number of BHPs, reduce. With less BHPs, there are less resources being requested, thereby lessening the resource contention in the core network.

Even with the availability of core resources, payload bursts may be dropped when they arrive at a node before resources have been allocated. This would normally be a result of diminished offset times due to long processing times. BHP processing times are typically low, about 1ns, but tend to increase with increase in number. Therefore, reducing the number of BHPs helps reduce the probability of burst drop.

7.2 Core Network Congestion Control

One of the advantages of OBS is in the great simplification of the core network. OBS is intended to have minimal management and therefore cheap to maintain. The immaturity of all optical processing, demands some electrical intelligence at LSRs.

We propose a BCN/FCN equivalent for optical networks to reduce complex signalling schemes used in SONET. We call this Optical-Congestion-Notification, OCN. Fig 4 shows a section of a meshed core network, and how OCN functions. While several feedback schemes have been proposed [53 54], OCN is a non-feedback scheme. No extra signalling packets are introduced in the OBS network.

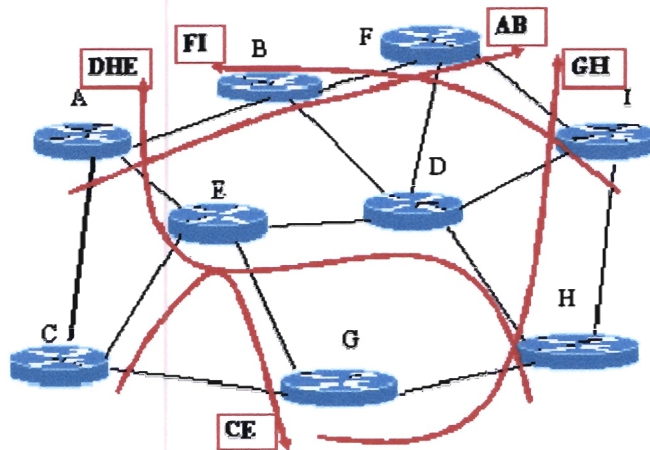


Fig. 7.2 The paths indicated, show routes taken by BHP packets. Passing each LSR, a BHP packet records the status of each LSR at that time. On reaching the next node, this information is read. The LSRs then use this information to route traffic accordingly.

The paths indicated, represent the different routes that may be taken by BHPs while reserving resources. Consider one of the above routes through nodes H, D, E and A. When the header packet reaches H, information of the available wavelengths at the time of processing is inserted into the header. On reaching D, this information is read, and information about D is also added to the header packet. The same process happens at E. At A, information about the route HDE is in the header packet.

The obvious setback of this is the accumulating information in the BHPs in a dense core network. To avoid this, header packets can be set to only carry LSR state information for a limited number of hops. This does not reduce the efficiency of OCN, since all other header packets in the core do congestion notification. There are two extreme cases that arise from OCN:

1. When there are many header packets to a node. When there are many header packets to a node, then more information about the neighbouring nodes is known. This then better helps in making routing decisions.
2. When there are very few header packets to a node. In this case, this then signals to the node that the neighbouring nodes are at low capacity. Bursts are then transmitted to the low capacity nodes.

The criteria for a node choosing or rejecting a route to a node, is based on available wavelengths and the delay that results in choosing a given route. This results in traffic in the core network being balanced. As a result, the efficiency of OBS is increased.

7.3 Burst Header Packet Fields

To enable core aggregation, and OCN extra fields in header packets must be added to those proposed in [29] must be added to header packets. Fig 5 shows the fields in a BHP packet without aggregation.



Fig. 7.3 The BHP shows fields appended at the ingress node, on entering the core network.

The descriptions of the individual fields are given below.

I_{pb} – Length of the payload burst. The burst lengths are determined at the ingress nodes. The BHP in this design does not have to wait for a burst to be assembled, for I_{pb} to be known. The length is known (approximated) using the H value, and method of FEC used. The time at which the BHP is transmitted is then the only modification in the JET protocol.

T_a – Time of arrival of the payload burst. Bandwidth is only reserved at the (expected) point of arrival.

T_d – The maximum delay tolerance of the payload burst. The value of T_d is real time and decreases with propagation delay. This value is used at both the ingress nodes and egress nodes. It helps in determining how long a burst should be held before transmission.

t_x – The QoS offset as described in [47]. The difference in this application, is that t_x is not fixed, but varies according to a given application, or FEC classification.

d_1, d_2 – Destinations to egress and final destination respectively. The need for the two separate fields rather than one is due FEC. During FEC payload bursts may be classified with respect to their egress nodes.

With core aggregation the information in the two individual header packets must be put be put in one BHP. Fig 6 shows a BHP packet that reserves resources for an aggregated burst.

l_{pb1}	l_{pb2}	T_{a1}	T_{a2}	T_{d1}	T_{d2}	t_{x1}	t_{x2}	D_c
-----------	-----------	----------	----------	----------	----------	----------	----------	-------

Fig. 7.4 The BHP shows fields appended at the in the core network by an LSR, the node at which burst aggregation occurs.

The descriptions of individual fields are given below.

$l_{pb1} l_{pb2}$ – The lengths of the aggregated bursts, 1 and 2.

$T_{a1} T_{a2}$ – The times of arrival (expected) of the two aggregated bursts. The value of δ is then obtained from the difference of the two.

$T_{d1} T_{d2}$ – The remaining tolerable delays for each of the bursts.

$t_{x1} t_{x2}$ – The QoS offsets for each of the bursts.

D_c – The core node for separation of aggregated burst.

The aggregated bursts in Fig3, separate at node D. The number of hops that aggregated bursts may transverse is determined by the egress destinations, and the availability of resources. Not shown in Fig 6, are the final burst destinations $d_1 d_2$. These remain the same as in Fig 5.

With the fields defined above, then core aggregation using LAUC-VF basic reservation scheme can be done as outlined below.

- 1 On arrival of header packets BHP1 and BHP2, the fields $T_a l_{pb} T_d$ and t_x are processed.
- 2 **If** $T_{a1} - T_{a2} \ll l_{pbmin}$

And destinations d_1 and d_2 transverse more than one node in the same direction, **then**

3 Aggregate bursts with destination D.

4 **else If** $T_{a1} - T_{a2} > I_{pbmin}$

Do not aggregate.

5 Go to step 1

University of Cape Town

Chapter 8

8 Conclusions and Recommendations

In this chapter, we give the conclusions of our findings and recommendations from our findings.

8.1 Conclusions

From results obtained from simulations in this study, we have shown the following:

1. Self-similar traffic has a significant effect on burst drops. There is more contention of resources in the core with self-similar traffic, compared to Poisson modelled traffic.
2. Self-similar traffic cannot justifiably be estimated to be close to Poisson modelled traffic. The change of traffic distribution over the Internet cannot be ignored, or assumed to be Poisson. Results from this study and other previous works do support this finding.
3. Assembled OBS traffic can be modelled as ON/OFF periods with ON periods being the burst lengths and OFF periods the offset periods. Modelling of OBS traffic as ON/OFF periods would help in mathematically modelling OBS traffic in relation incoming to self-similar traffic.
4. That randomness of offset assignment has a direct relation to contention of resources at LSRs. The more random an offset assignment algorithm, within minimum and maximum limits, the more random the arrival of bursts at LSRs. Random burst arrival at LSRs reduce contention.
5. That the proposed FEC assembly scheme results in shorter packet queue length and longer burst lengths which results in less contention of resources. The proposed FEC-Based assembly scheme is faster. Combining the FEC-Based scheme with Pareto offset assignment, even less contention is observed.

We conclude that there are three interdependent factors that affect OBS network performance:

1. Packet queue management
2. Method used to assemble the packets
3. Offset assignment scheme used

8.2 Recommendations

From this study, we make several recommendations.

1. Future studies should implement a queue network such as one used in this study, a queue for the packets and bursts.
2. Self-similar traffic should be used for simulations to give a realistic performance of the OBS network
3. Constant offset assignment must not be used for offset assignment
4. A Pareto offset assignment be used to assign offset periods to bursts
5. Field Equivalence classification should be used to classify packets and bursts at the LER

During the simulations we could not carry out long duration of simulations due to lack of memory on the simulation machine used. This was due to the fact that running of the simulations meant creating several hundreds of megabytes of data per simulation. Long duration of simulations resulted in the simulating machine crashing. We therefore had to find an appropriate simulation time that would allow for accurate simulation results. From the simulation results, the trend of the data was forecast using logarithmic trend line. More accurate results may have been got with higher performance machines.

Though C++ was the primary programming language for the simulations, sequential programming was used and not parallel programming. We do not think using sequential rather

than parallel programming would affect the conclusions from our results, but rather, the algorithms would have certainly been faster for all simulated algorithms. All delays may all be lower than those presented in this work.

University of Cape Town

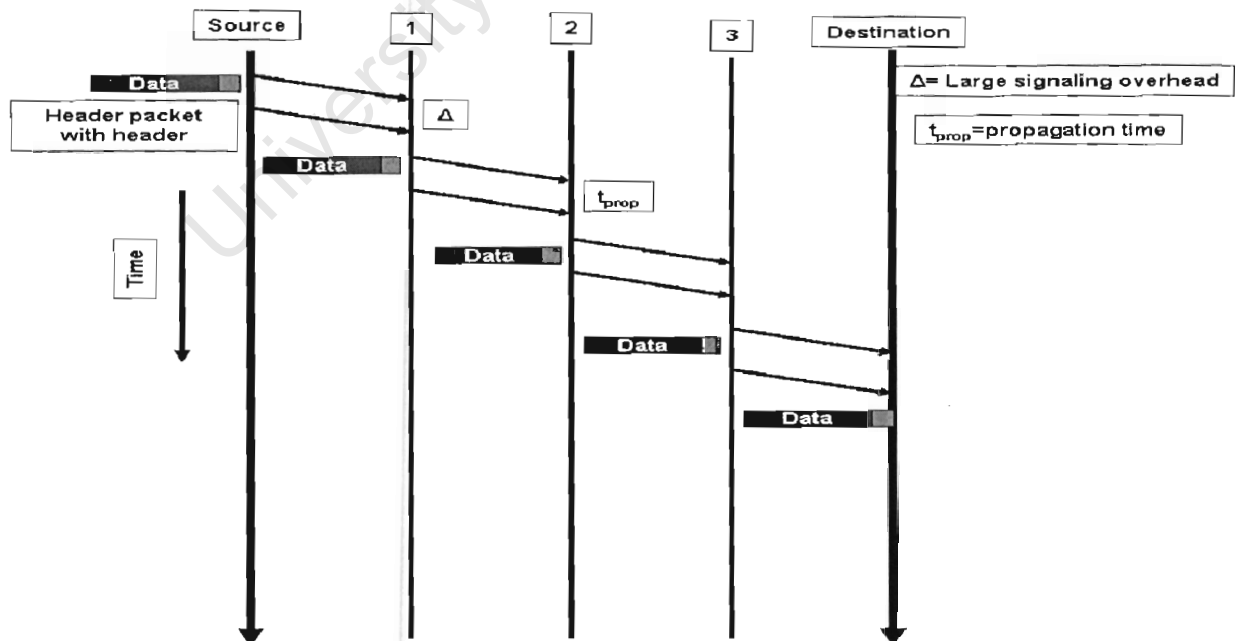
Appendix A: The OBS-0.9a Simulator

The OBS-0.9a software is an open source software running on ns2. It simulates the very basic Optical Burst switching protocol functionalities. OBS-0.9a can be downloaded for free from [52]. An updated version with further enhancements has been made available too in the last few months (early 2006).

Appendix B: Timing Relationships for Circuit and Packet Switching

Packet switching

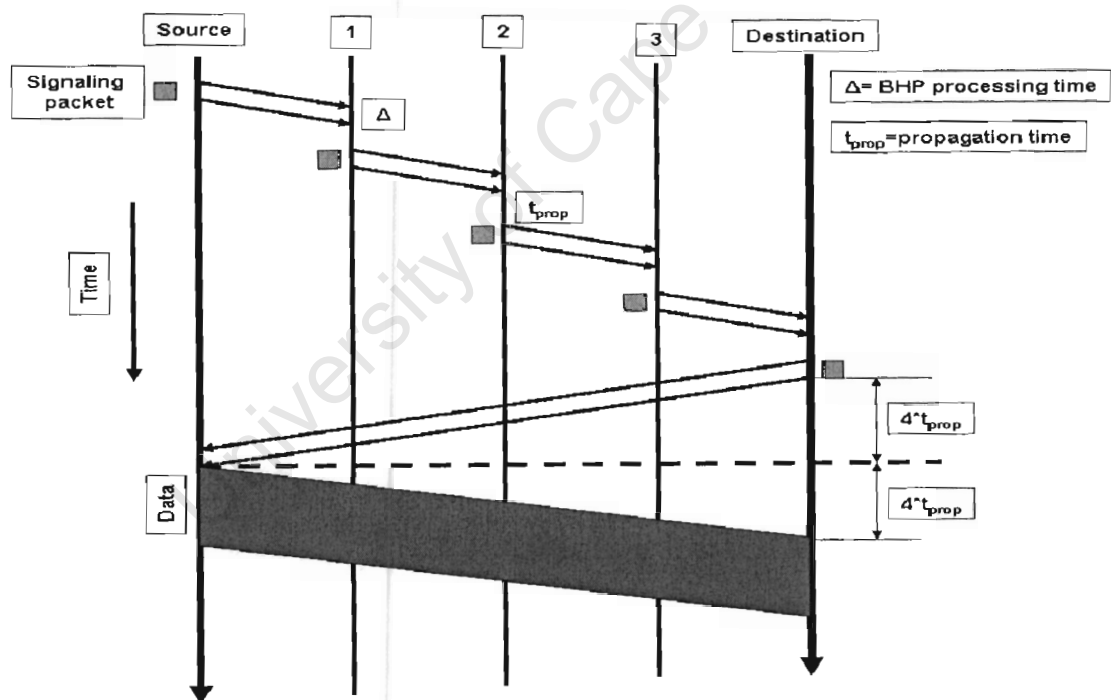
In packet switching, the payload is transmitted with the header packet. No signalling is done for individual packets, resources are therefore not reserved before hand. Ideally, real packet switching is only feasible in a two layered network model, where the packet is processed optically without the need for buffering. Without all optical processing capabilities, core nodes become a bottle neck, limiting the amount of data that can be transmitted in a core network. No QoS can be guaranteed.



The main advantage of packet switching is the short delay that is incurred by individual packets. The downside of no signalling in packet switching is the large signalling that is needed to “guide” the packet through the network

Circuit Switching

In circuit switching, a path is set up before sending the payload. The setting up of a path that guarantees payload QoS demands and no processing of the payload is required since nodes are already configured from the time of transmission of the payload from the ingress node. The downside of circuit switching is the delay that must be incurred by the payload before transmission for the path to be set-up, and for receiving of all acknowledgements from each node for a successful resource reservation.



Appendix C: Introduction to Self-Similarity

This chapter introduces self-similarity as a distribution of Internet traffic. We discuss the implications of the self-similarity nature of traffic to the performance of the core network.

Understanding Self-Similarity

To do any kind of core network design that involves traffic, an accurate traffic model must be used. The assumptions made to achieve the model must not deviate greatly from the real life scenario. This chapter shows the model of Internet traffic used in this study.

Modelling of traffic in the core network started with the modelling of telephonic traffic. Voice packets from one source have a constant mean and therefore a predictable arrival rate at a given node. Voice traffic therefore used Poisson equation to model voice communication. However, in recent years, the distribution of Internet traffic has deviated from the Poisson model to a model that depict bursty periods on all time scales. This study was first done in [21], and numerous studies have followed supporting the results.

It is now accepted that the bursty periods are observed on a wide range of timescales, and can be assumed to exist on all timescales. This existence of bursty periods on all timescales is said to be self-similar. The self-similar nature of Internet traffic implies that, on the time-scale, the distribution in a one hour interval is similar to a distribution in a ten day interval.

Definition.B1: Self-similarity is the existence of random bursty periods, of five times or more from the mean, on all timescales.

The evidence of self-similarity has helped in understanding the behaviours observed in Internet traffic. Most importantly though, the understanding that Internet traffic is much more bursty than the Poisson model is helping traffic engineers design buffers and queuing models more optimally. The cause of self-similarity has largely been due to the enormous increase in Internet traffic coupled with the rapidly increasing different application protocols, each of which have different transmission rates.

With the understanding of the self-similar nature of traffic, a mathematical formulation must be done if any modelling is to be done. The standard method is to plot the collected data, and then from available distributions, choose a goodness fit. Results in [21] show that Internet traffic can be modelled using heavy-tailed distributions. The existence of heavy tails in Internet traffic is contrary to the Poisson model that exhibits light-tails.

Definition.B2: Heavy-tail distributions, also known as Power-Law distributions, are those distributions that are not memoryless, compared to light-tailed distributions that are memoryless. A formal mathematical definition is that for a random variable X ;

$$P[X > x] \sim x^{-\alpha}, \text{ as } x \rightarrow \infty, 0 < \alpha < 2$$

There are several heavy-tailed distributions. These are the: Pareto distributions, Log-normal distributions, Cauchy distributions, and the Weibull distributions. The easiest of the distributions, is the Pareto distribution. The mathematics involved is relatively less cumbersome compared to other heavy-tailed distributions. The Pareto distribution can be, to a very good approximation, computed during simulations.

In this study, we use the Pareto distribution to generate Internet traffic. We follow the assumptions that are widely accepted in traffic engineering.

In the next section, we discuss in detail the Pareto distribution, and outline the limitations encountered when using it. Where appropriate, in the following sections, we give plots from simulations, of Pareto distributed traffic, and compare these plots with those of the traditional Exponential distribution. We repeat the simulations in [21] that conclusively show that aggregation of multiple of Pareto distributions result in bursty traffic that closely compares to real Internet traffic. We also show that using Exponential distributions, the burstiness of the traffic greatly underestimates the burstiness of real Internet traffic.

The Pareto Distribution and Self-Similarity

This section describes the Pareto distribution and how it is used to model Internet traffic. The Pareto distribution has been applied extensively in analysing economic data distributions. It is also used to model the distribution of populations in a country. In communication, the Pareto distribution has been used to model call holding times in [3], and variable-bit-rates frame sizes in [4]. In recent years much research has been done to understand its application in communication networks, Internet traffic, and Queuing models.

The Pareto distribution has been shown to adequately model the Self-Similarity exhibited in Internet traffic [20]. Equation 1 shows the Pareto cumulative density function. The Pareto distribution has three parameters that describe it. The distribution can be described in terms of:

1. Scale using the scale parameter β
2. Shape of the tail of the distribution using the α parameter
3. Location or size of the smallest burst length using the b parameter

$$F(x) = 1 - \left(\frac{\beta}{x + \beta - b} \right)^\alpha \text{ With } (x \geq b) \quad (1)$$

where b , α and β are always greater or equal to zero ($b, \alpha, \beta \geq 0$).

Assuming self-similarity, we can equate b to β . This means that self-similarity is to the scale of the minimum burst length b . Equating the shape parameter to the locate parameter also significantly simplifies the mathematical manipulation and understanding. Then Equation one becomes:

$$F(x) = 1 - \left(\frac{b}{x} \right)^\alpha \text{ With } (\beta = b) \quad (2)$$

The same limitations of equation 1 apply to equation 2. The derivative of equation 2 gives the probability density distribution. For the derivation of equation three from equation 1 see Appendix C.

$$f(x) = \frac{\alpha b^\alpha}{x^{1+\alpha}} \quad (3)$$

However, for the above equations to adequately model the burstiness that is observed in Internet traffic, we must understand the behaviour of the mean and variance of both the Pareto distribution and Internet traffic. Internet traffic has a wide range of variance that can be approximated to be infinite, and a finite mean distribution. The Pareto distribution has the following mean and variance characteristics. For the Pareto distribution, let X be a random variable. Then Table I shows the characteristics of the random variable X with different values of

the shape parameter. From Table I therefore, the shape parameter should be between one and two.

Table 10 Pareto traffic parameters

SHAPE	SCALE	MEAN	VARIANCE
$0 \leq \alpha \leq 1$	$\beta = b$	INFINITE MEAN	FINITE VARIANCE
$1 < \alpha \leq 2$	$\beta = b$	FINITE MEAN	INFINITE VARIANCE

To put into context the Pareto distribution as described in equation (3), we describe each of the equation in terms of packets. Consider a traffic stream of packets with length distributions of x , then b is the minimum length of the packets. The variable α is used to describe the variance of the distribution of the packet lengths. The shape parameter α is determined in relation to a quantitative measure of the Hurst parameter H , which can be calculated, given a set of data. For self-similarity, the Hurst parameter varies between $1/2 < H < 1$. Self-similarity increases as $H \rightarrow 1$. The equation relating H to α is,

$$H = (3 - \alpha) / 2 \quad (4)$$

Knowing the mathematical characteristics however, does not give a practical understanding of how the Pareto distribution with a finite mean and infinite mean model Internet traffic. The proposed method of aggregated periods of ON/OFF in [21] are used in this study.

Appendix D: Deriving the First Moment of Pareto Equation

$$F(x) = 1 - \left(\frac{b}{x}\right)^\alpha \quad \text{With } (\beta = b)$$

$$f(x) = \frac{d}{dx}(F(x))$$

$$\frac{d}{dy} F(x) = \frac{d}{dx}(1) - \frac{d}{dx}(\gamma^\alpha x^{-\alpha})$$

$$f(x) = -\frac{d}{dx}(\gamma^\alpha x^{-\alpha})$$

$$f(x) = -\gamma^\alpha(-\alpha x^{-\alpha-1})$$

$$f(x) = \frac{\alpha \gamma^\alpha}{x^{\alpha+1}}$$

Appendix E: Queuing Theory Terminology

In this Appendix we define certain concepts in queues, but in the context of events that take place at the LER.

There are two sets of queues at the LER, the packet queue and the burst queue. The service times of the packets during assembly are a critical factor at the buffer. Slow service rates or extended periods of service of the packets would lead to buffer overflow and unwarranted delays that violate the delay tolerances of the individual packets.

We will make use of the well known Kendall's notation A/B/C/D/E/F give a brief description of each term and how it relates to events at the LER.

A is the probability distribution of the arrival of the packets. In the context of events at the LER, there are two arrival processes, the arrival of packets that are to be transmitted into the core network, and the burstification rate of the bursts. These two rates of arrivals have different effects on their respective queues. We label packet arrivals as λ_1 and the rate of burstification as λ_2 . It should be noted that the rate of arrivals are not Poisson.

B is the probability distribution of the service rate. The service rate in this study is broken down into two parts: The service rate of the packets and the service rate of the bursts. The service rate of the packets is the rate at which the packets start being assembled. The service rate of the bursts is the rate at which the bursts are being transmitted into the core network. We label the packet service rate as μ_1 and the burst service rate as μ_2 .

C is the number of channels or servers that are connected to the node. In the context of the LER, C is the number of wavelengths that are available to the LER.

D is the maximum number of packets or bursts in the respective queues. This refers to packets or bursts that are both in the waiting to be served, and those that are being served. We label the maximum packet and burst maximum queue lengths as L_{m1} and L_{m2} respectively.

E is the total number of packets and bursts in the LER. This defines the size of the buffer and will be the sum of L_{m1} and L_{m2} .

F is the queue discipline being used. We implement a fair queuing discipline for both packets and bursts.

Appendix F: Introduction to Decision Theory

Decision theory is used to understand, mathematically, how decisions are made. It is from decision theory that preference mathematics emanates. Consider any two objects **a** and **b** one of which must be chosen. $a \succ b$ states the decision maker strictly prefers object a to object b. If there are three objects, a b and c, then $a \succ b$ and $b \succ c$ means the decision maker strictly prefers a to b. However, should a not be available, then preference would be for object c. Therefore $a \succ c$ holds. The preference is therefore transitive. The preference can also be shown to be asymmetric. Consider objects a and b. If $a \succ b$ then $b \succ a$ cannot hold for strict preference.

There is also a case of indifference represented by \sim . Therefore, $a \sim b$ means the decision maker does not mind either object a or b. Other obvious cases that hold are $a \sim a$ and $b \sim b$. For three objects a b and c, if $a \sim b$ and $b \sim c$ hold, then $a \sim c$ holds as well.

When a decision maker prefers one object over another but does not mind having the other object this is called relative preference. We shall use $\succ \sim$ to denote relative preference. For objects a and b, then $a \succ \sim b$ would mean the decision maker prefers object a to object b, however, should a not be available, then the decision maker will choose object b. The difference between strict preference and relative preference is the flexibility of choice introduced.

Appendix G: Random Offset Assignment Distributions

We use random number generators to generate the random numbers according to the different distributions. The random number generator uses the default uniform generator in the Linux operating system. The *myrand* function below returns a uniformly distributed number between 0 and 1. The calling the *srand* function allows for different random numbers for every simulation.

```
inline double myrand (void) {
    double r;

    //providing a different seed every time for the random generator.
    srand((unsigned)time(0));

    /* random value in range [0,1] */
    r = ( (double)rand() / (double)(RAND_MAX) );

    return(r);
}
```

Uniform distribution

The uniform distribution function (*myrand*) is limited within two extreme values that can be varied accordingly with a specified mean. *Min_off* corresponds to the minimum offset time, *max_off* corresponds to the maximum offset assigned to a burst and *mean_off* corresponds to the mean of the offset. Therefore uniform (0.0001, 0.0005, 0.00025) would give a Uniformly distributed values within the minimum offset of 1ms and a maximum offset of 5ms with a mean of 2.5ms. The generated value is then assigned as an offset to a burst.

```
inline int uniform (int min_off,int max_off, int mean_off) {

    //Uniform distribution based on average and min
    int y=max_off+1;
    while(y>max_off){
        y=min_off +(int)(2*(mean_off-min_off)*myrand());
    }
    //offset=y;
    return(y);
}
```

Pareto Distribution

The Pareto number generator uses the power law definition. For Simulation, we use the

formula $X_{Pareto} = \frac{b}{(myrand)^k}$, where b is the minimum value that the minimum offset can take. Calling the Pareto function, Pareto (1.85, 0.0002, 0.0005) would give a Pareto distributed random number. The generated value would then be assigned as an offset to a burst.

```
inline double Pareto (double k,double min_off,int max_off) {  
// Generate Pareto random numbers  
// k between 1 and 2, closer to 1 means more long periods  
// min_off is the minimum value for the period  
  
double w=min_off/pow(1-myrand(),1/k);  
// mean=1st moment=min_off*k/(k-1) => k= mean/(mean-min_off)  
// median = min_off*(2**(1/k)), this is 50th percentile  
  
while(w>=max_off){  
w=m/pow(1-myrand(),1/k); //This is based on F^-1, not f^-1!  
}  
return(w);  
  
}
```

References

- [1] C. Qiao, M. Yoo, "Choices Features and Issues in Optical Burst Switching", *Optical Networks Magazine*, 1(2), pp. 36-44, May 2000.
- [2] S. Verma, H. Chaskar and R. Ravikanth, "Optical burst switching: a viable solution for terabit IP backbone," *IEEE Network*, vol. 14, no. 6, pp. 48-53, 2000.
- [3] XILINX SONET/SDH. http://www.xilinx.com/esp/wired/optical/collateral/sonet_sdh.pdf, Accessed February 2006
- [4] G. Rosenfeld, "The Technology and Economics of RPR – Part II. RPR and MPLS Over existing SONET Networks- Why should I care? ", *Analogue Zone*
- [5] Angilent Technologies, "Next Generation SONET/SDH devices – the driver for multi-port, multi-channel test", <http://cp.literature.agilent.com/litweb/pdf/5988-8265EN.pdf>, Accessed March 2006
- [6] G. Goebel "The British Invention Of Radar", <http://www.vectorsite.net/ttwiz1.html>. Accessed March 2006
- [7] IEC Tutorial, "Dense Wavelength Division Multiplexing (DWDM)", <http://www.iec.org/online/tutorials/dwdm/>. Accessed on March 2005
- [8] Nec Sets New 10.9T/S Ultra-Dense WDM Transmission Capacity World Record, www.nec.co.jp/press/en/01032201.html. Accessed January 18, 2006
- [9] C. Qiao and M. Yoo, "Optical burst switching - a new paradigm for an optical Internet," *J. High-Speed Networks*, vol. 8, no. 1, pp. 79-90, 1999.
- [10] K. Sriram, D.W. Griffith, SuKyoung Lee, N.T. Golmie, "Optical Burst Switching: Benefits and Challenge", *Proceedings of the First International Workshop on Optical Burst Switching (WOBS 2003)*, Dallas/TX, October 2003
- [11] J. Xu, H. Perros, G. Rouskas, "Techniques for Optical Packet Switching and Optical Burst Switching", *IEEE Communication Magazine*, Vol. 39, no. 1, pp. 136-142, January 2001.
- [12] F. Callegati, H.C. Cankaya, Yijun Xiong, and M. Vandenhouete, "Design issues of optical IP routers for Internet backbone applications", *IEEE Communications Magazine*, vol. 37, issue 12, pg. 124-128, Dec. 1999.
- [13] C. Qiao and M. Yoo, "Choices, Features and Issues in Optical Burst Switching", *Optical Networks Magazine*, Vol. 1, No. 2, pp. 36-44, 2000.
- [14] W. E. Leland, M. S Taqqu, W. Willinger, and D. V. Wilson. "On the self-similar nature of Ethernet traffic." *A CM/SJGCOMM Computer Communications Reuzew*, Vol. 23, pp. 1S3-193, 1993. *Proceedings of the ACM/SIGCOMM'93*, San Francisco, September 1993. September 1993.
- [15] M. Dueser and P. Bayvel, "Analysis of a dynamically wavelength-routed optical burst switched network architecture." *Journal of Lightwave Technology*, No 20, pp.574-585, April 2002.
- [16] M. Dueser and P. Bayvel, "Performance of a dynamically wavelength-routed optical burst switched network." *IEEE Photonics Technology Letters*, Vol 14(2), pp239-241, Feb 2002.
- [17] X. Yu, Y. Chen, and C. Qiao, "Study of traffic statistics of assembled burst traffic in optical burst switched networks," in *Proceeding of Opticomm*, 2002, pp. 149-159.
- [18] A. Ge, F. Callegati, and L. Tamil, "On optical burst switching and Self-similar traffic," *IEEE Communications Letters*, vol. 4, pp. 98-100, March 2000.
- [19] F. Xue, S. J. B. Yoo, "Self-similar traffic shaping at the edge router in optical packet switched networks." *Proceedings of IEEE International Conference on Communications (ICC 2002)*, 2002.
- [20] W. E, Leland, M. S. Taqqu, W. Willinger, and D. V, Wilson. "On the self-similar nature of Ethernet traffic (Extended Version)." *IEEE/ACM Transactions on Networking*, Vol. 2, pp. 1-15, 1994.
- [21] V. Paxson and S. Floyd, "Wide-area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, pp.226-244, June 1995.
- [22] K. C. Sevcik and I. Mitrani, "The Distribution of Queuing Network States at Input and Output Instants." *Journal of the Association for Computing Machinery*, Vol 28, No 2, April 1981, pp 358-371.
- [23] G. Hu, K. Dolzer, C.M. Gauger: "Does burst assembly really reduce the self-similarity?" *Proceedings of the Optical Fiber Communication Conference (OFC 2003)*, Atlanta, March 2003.
- [24] X. Yu, Y. Chen, and C. Qiao. "Study of traffic statistics of assembled burst traffic in optical burst switched networks." In *Proceeding of Opticomm*, pages 149-159, 2002.
- [25] A. Ge, F. Callegati, and L. S. Tamil "On Optical Burst Switching and Self-Similar Traffic", *IEEE Communication Letters* March 2000.

- [26] S Junghans, and C. M. Gauger, "Architectures for Resource Reservation Modules for Optical Burst Switching Core Nodes." ITG-Fachtagung Photonics Networks, May 5-6, 2003.
- [27] J. Ramamirtham and J. Turner, "Design of Wavelength Converting Switches for Optical Burst Switching." WUCS-01-21, August 7 2001.
- [28] C. Cameron, A. Zalesky and M. Zukerman, "Prioritized Deflection Routing in Optical Burst Switching Networks." IEICE Transactions on Communication, VolE88-B, No 5, pp1861-1867 Dec 2004.
- [29] Y. Xiong, M. Vandenhouste and H. C. Cankaya, "Control Architecture in Optical Burst-switched WDM Networks," IEEE J. Select. Areas Commun., vol. 18, no. 10, pp. 1838-1851, 2000.
- [30] J. S. Turner, "Terabit Burst Switching," J. High-Speed Networks, vol. 8, no. 1, pp. 3-16, 1999.
- [31] J. Xu, C. Qiao, J. Li, and G. Xu, "Efficient Channel Scheduling Algorithms in Optical Burst Switched Networks," in Proceedings of INFOCOMM, 2003, vol. 3, pp. 2268–2278.
- [32] H. Li, H. Neo, and T. Ian "Performance of The Implementation of A PipeLine Buffering System In Optical Burst Switching Networks." ISBN 0-7803-7975-6-03 2003.
- [33] J. Li, et al, "Maximizing Throughput for Optical Burst Switching Networks." IEEE INFOCOM 2004.
- [34] G. C. Hudek and D. and J. Muder, "Signaling analysis for a multi-switch all-optical network" 1995 IEEE international conference in communications pp 1206-1210, June 1995.
- [35] Hailong Li, Hanmeng Neo, and Thng Li-Jin Ian "Performance of The Implementation of A PipeLine Buffering System In Optical Burst Switching Networks" Globecom2003.
- [36] J. Y. Wei and R. I. McFarland, "Just-In-Time signaling for WDM optical burst switching networks," IEEE/OSA Journal of Lightwave Technology, vol. 18, no. 12, pp. 2019–2037, December 2000.
- [37] M. Yoo and C. Qiao, "Just-enough-time (JET): A high speed protocol for bursty traffic in optical networks," in Proceeding of IEEE/LEOS Conf. on Technologies for a Global Information Infrastructure, August 1997, pp. 26–27.
- [38] J. Ramamirtham and J. Turner, "Time Sliced Optical Burst Switching." Proceedings of IEEE INFOCOM, April 2003.
- [39] H. Øverby and N. Stol, "Providing absolute QoS in asynchronous bufferless optical packet/burst switched networks with the adaptive preemptive drop policy." Computer Communications Volume 28, Issue 9 , pp 1038-1049, 2 June 2005
- [40] G. Hu, et al, "Traffic and performance analysis of optical packet/burst assembly with self similar traffic." Proceedings of the 7th International Conference on Transparent Optical Networks (ICTON 2005), Barcelona, 2005.
- [41] J. Kögel, et al, "Design and Evaluation of a Burst Assembly Unit for Optical Burst Switching on a Network Processor. EUNICE 2005: Networks and Applications Towards a Ubiquitously Connected World, Springer, 2005.
- [42] X. Cao, J. Li, Y. Chen, and C. Qiao, "Assembling TCP/IP Packets in Optical Burst Switched Networks", Proceedings, IEEE Globecom 2002.
- [43] 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, No.7, September 2003.
- [44] M. Yoo, C. Qiao and Sudhir Dixit, "QoS performance of optical burst Switching in IP-over-WDM networks", IEEE J. Select. Areas Commun., vol. 18, No. 10, pp. 2062-2070, Oct. 2000.
- [45] K. Long, R. S. Tucker and C. Wang "A New Framework and Burst Assembly for IP DiffServ over Optical Burst Switching Networks" Elsevier Science Direct Journal, August 2005.
- [46] L. Xinwan et al "An Experimental Study of an Optical Burst Switching Network Based on Wavelength-selective Optical Switches", IEEE Communications Magazine, pp S3-S11, May 2005.
- [47] M. Yoo and C. Qiao, "A new optical burst switching (OBS) protocol for supporting quality of service," iSPIE Proc., "All Optical Commun. Syst.: Architecture, Control Network Issues", vol. 3531, pp. 396–405 Nov. 1998.
- [48] V. M Vokkarane, K. Haridos and J. P. Jue, "Threshold-Based Burst Assembly Policies for QoS Support in Optical Burst-Switched Networks" Proceedings, Optical Networking and Communication Conference (OptiComm) 2002, Boston, MA, July-Aug 2002.
- [49] K. Dolzer and C. Gauger, "On burst assembly in optical burst switching networks - a performance evaluation of Just-Enough-Time", Proceedings of the 17th International Teletraffic Congress (ITC 17), Salvador, September 2001.
- [50] S. French "Decision Theory: an introduction to the mathematics of rationality." Ellis Horwood Limited 1988.
- [51] www.isi.edu/nsnam/ns/

- [52] www.dawn.cs.umbc.edu/owns/
- [53] F.Xue, "End-to-end Contention Resolution Schemes for an Optical Packet Switching Network With Enhanced Edge Routers," *Journal of Lightwave Technology*, Vol.21, no.11, pp. 1-10, November 2003.
- [54] A.Maach, G.v.Bochmann, H.Mouftah, "Contention avoidance in optical burst switching," pp.1-7, 2004

University of Cape Town