

國立交通大學
資訊科學與工程研究所
博士論文

全光分波多工封包交換都會環狀網路之
高效能服務品質保證媒介存取控制技術

A High-Performance Medium Access Control
Scheme with QoS Assurance for an Optical
Packet-Switched WDM Metro Ring Network



研 究 生：趙一芬

指 導 教 授：楊啟瑞 博士

中 華 民 國 九 十 九 年 四 月

全光分波多工封包交換都會環狀網路之
高效能服務品質保證媒介存取控制技術
A High-Performance Medium Access Control
Scheme with QoS Assurance for an Optical
Packet-Switched WDM Metro Ring Network

研究生：趙一芬

Student: I-Fen Chao

指導教授：楊啟瑞 博士

Advisor: Dr. Maria C. Yuang



A Thesis
Submitted to Department of Computer Science
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
For the Degree of
Doctor of Philosophy
in
Computer Science
March 2010
Hsinchu, Taiwan, R.O.C.

中華民國九十九年四月

全光分波多工封包交換都會環狀網路之 高效能服務品質保證媒介存取控制技術

研究生：趙一芬

指導教授：楊啟瑞 博士

國立交通大學 資訊工程學系

Abstract in Chinese

下一代全光都會型網路(Metropolitan Area Networks; MANs)旨在低成本有效率地運用先進的光封包交換技術(Optical Packet Switching; OPS)支援各類型要求高頻寬之網路應用程式以及訊務特性趨於動態變化之網路應用程式。此篇論文提出一高效能服務品質保證媒介存取控制機制，應用在我們建立的高效能分波多工光封包交換都會環狀網路(High-performance OPS Metro WDM slotted-ring Network, HOPSMAN)實驗平台。HOPSMAN 的設計為一可擴展性架構，所以網路節點數目可不受光波道的數目限制。HOPSMAN 網路中包含數個服務節點，額外配備時槽除訊器，具備時槽除訊功能，以高效率低成本的方式增加頻寬利用率。HOPSMAN 最重要的設計為其獨一無二的媒介存取控制機制，稱為機率式定額與額外配額(Probabilistic Quota plus Credit; PQOC)；而後，我們又在其上加入服務品質保證(Quality of Service; QoS)的功能，稱為機率式定額與額外配額服務品質保證(Probabilistic Quota plus Credit with QoS Assurance; PQOC/QA)。藉由機率式定額傳送資料的方式，本媒介存取控制機制可以高效率且公平的使用頻寬。根據服務節點的數量，目標節點的訊務量分配方式，我們分析計算出該定額分配量。除此之外，為應付極具動態變化的都會型網路的訊務量，本媒介存取控制機制引進一時間控制機制的額外配額方式來公平使用多餘的頻寬。更甚者，為了支援服務品質保證以及解決分波多工網路固有的存取問題，PQOC/QA 利用簡單且具彈性的標記方法來執行時槽預訂。為了更適應動態的即時訊務(VBR)，我

們不像以往其它的研究著重於估算動態訊務量；取而代之，PQOC/QA 簡單地採用平均連線速率頻寬保留的方法在環狀網路上的每個循環(cycle)預定保留頻寬，彈性地建立即時連線以傳送動態的即時訊務。另外，根據 M/G/m 排隊理論的分析，我們發展了一個獨特的概算方式求得平均建立連線等待時間。本分析的伺服器數量為系統預先定義的即時訊務的最大可接受定額數，服務時間包含一指數分佈之長度及外加一常數值。關於 M/G/m 排隊理論，除了少數的服務分佈可以得到準確結果之外；以往，針對具某些特性的一般服務分佈的概算分析，只能達到 10%的相對誤差值。但我們針對本系統內特定的服務分佈，我們提出一概算方式，所求得的结果與模擬實驗結果完全吻合。更甚者，經由深入的模擬結果，藉由本篇論文所提出的媒介存取控制機制，即使在各種負載或大量突發訊務之下，HOPSMAN 可以達到更為優異的系統輸出，低延遲，以及卓越的即時訊務表現，達到高即時訊務輸出，以及極低的 VBR 延遲及延遲變化量。



A High-Performance Medium Access Control Scheme with QoS Assurance for an Optical Packet-Switched WDM Metro Ring Network

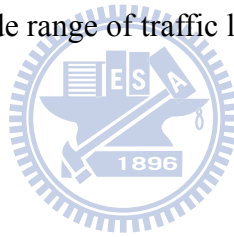
Student: I-Fen Chao *Advisor:* Dr. Maria C. Yuang

Department of Computer Science
National Chiao Tung University, Taiwan

Abstract

Future optical Metropolitan Area Networks (MANs) have been expected to exploit advanced Optical Packet Switching (OPS) technologies to cost-effectively satisfy a wide range of applications having time-varying and high bandwidth demands and stringent delay requirements. In this thesis, we present a high-performance real-time medium access control scheme for our experimental high-performance OPS metro WDM slotted-ring network (HOPSMAN). HOPSMAN has a scalable architecture in which the node number is unconstrained by the wavelength number. It encompasses a handful of nodes (called server nodes) that are additionally equipped with optical slot erasers capable of erasing optical slots resulting in an increase in bandwidth efficiency. In essence, HOPSMAN is governed by a novel medium access control (MAC) scheme, called Probabilistic Quota plus Credit (PQOC), which is further enhanced with QoS assurance, called Probabilistic Quota plus Credit with QoS Assurance (PQOC/QA). The proposed MAC scheme embodies a highly efficient and fair bandwidth allocation in accordance with a quota being exerted probabilistically. The probabilistic quota is then analytically derived taking the server-node number and destination-traffic distribution into account. Besides, the MAC scheme introduces a time-controlled credit for regulating a fair use of remaining bandwidth particularly in the metro environment with traffic of high burstiness. Moreover, PQOC/QA adopts slot-basis reservation through a simple and flexible marking mechanism to support QoS and to resolve the intrinsic access problem in WDM network. Instead of focusing on estimation of the bandwidth requirements, PQOC/QA sets up real-time connections by employing constant mean rate reservation on each cycle of the ring

and effectively accommodates bursty real-time traffic (VBR). Furthermore, we develop a novel approximation to acquire the accurate results of the expected connection setup queueing delay by means of an M/G/m queueing analysis. In the analysis, the maximum admissible quota of real-time traffic is regarded as the number of servers and the service time has a duration that follows an exponential form with an added constant. In M/G/m queueing analysis, the accurate results have only been attained for a limited number of special service distributions, while most of the proposed approximation only maintained a less than 10% relative error for certain properties of service distributions. Our approximation results, which are derived under the particular general service distribution in our system, show that the mean setup queueing time is in profound agreement with the analytic result. Additionally, extensive simulation results show that HOPSMAN with the proposed MAC scheme achieves exceptional delay-throughput performance and remarkable real-time traffic performance (high statistical multiplexing gain for real-time traffic, exceedingly low VBR delay and jitter) under a wide range of traffic loads and burstiness.



誌謝

首先我要對我的指導老師楊啟瑞教授致上最誠摯的謝意與敬意。感謝她在我博士班期間不厭其煩地給我指導與協助，以及研究理念上的薰陶。同時也在老師身上看到了她對做研究的熱忱和堅持，這樣的精神讓我深受感動以及深刻地影響我。

接著，我要特別感謝我的先生 鍾勇輝，沒有他全力的支持和鼓勵，我想我不會走完這一程。每每氣餒時，他總是對我仍然信心十足，讓我有勇氣去面對每個時候的難題。

我還要感謝實驗室的學長學弟，他們在這幾年間給予我許多的指導與關照。感謝羅志鵬、施汝霖先生在研究過程中不吝給予幫助指導，感謝與王雅織學姐互相砥礪扶持。

此外，我還要感謝我的家人，因為你們的支持與鼓勵，使我有動力完成這份研究，在此衷心地感謝他們。特別是我婆婆，由於她無私的照料，讓我完全無後顧之憂，可以全心的專注在研究上。感謝我的女兒鍾若昀，讓我在研究之餘，感覺生命的美好。

最後僅將我的論文獻給我最摯愛的父母，感謝他們無條件的愛以及支持。

趙一芬

國立交通大學
中華民國九十九年五月

CONTENTS

ABSTRACT IN CHINESE	I
ABSTRACT.....	III
ACRONYMS.....	IX
CHAPTER 1. INTRODUCTION.....	1
1.1 OPTICAL WDM NETWORKS	1
1.2 MOTIVATION AND OBJECTIVES	11
1.3 ORGANIZATION OF THE THESIS.....	13
CHAPTER 2. GENERAL NETWORK AND NODE ARCHITECTURES	15
2.1 NETWORK ARCHITECTURE	15
2.2 NODE ARCHITECTURE	16
CHAPTER 3. MAC SCHEME– PROBABILISTIC QUOTA PLUS CREDIT ...	19
3.1 DESIGN PRINCIPLES AND THE DETAILED ALGORITHM	19
3.2 BANDWIDTH ALLOCATION- PROBABILISTIC QUOTA DETERMINATION	25
3.3 SIMULATION RESULTS	30
3.4 TESTBED IMPLEMENTATION AND EXPERIMENTAL RESULTS	40
CHAPTER 4. MAC SCHEME WITH QOS ASSURANCE	50
4.1 DESIGN PRINCIPLES AND THE DETAILED ALGORITHM	50
4.2 EXPECTED QUEUEING TIME ANALYSIS.....	64
4.3 SIMULATION RESULTS	71
CHAPTER 5. CONCLUSIONS.....	84
REFERENCES.....	86

List of Figures

Figure 1. HOPSMAN: network architecture	15
Figure 2. HOPSMAN node architecture.	16
Figure 3. Cycle and slot structures	21
Figure 4. Detailed PQOC algorithm.....	23
Figure 5. Quota determination.....	26
Figure 6. Bandwidth efficiency of HOPSMAN	32
Figure 7. Analytic and simulation results on system throughput under different S-node	33
Figure 8. Throughput performance of HOPSMAN.....	34
Figure 9. Delay performance of HOPSMAN.....	35
Figure 10. Credit window size under various burstiness.....	37
Figure 11. Credit impact on delay for network with malicious nodes (nodes 5 and 15)	37
Figure 12. The impact of probabilistic exertion of quota under various loads and burstiness.....	38
Figure 13. Hardware implementation of the HOPSMAN testbed system.	41
Figure 14. Synchronization of control and data channels	43
Figure 15. Experimental results with fast optical devices.....	46
Figure 16. Feasibility test and demonstration of HOPSMAN testbed	48
Figure 17. Cycle and slot and reservation structures.....	51
Figure 18. Quota distribution in PQOC/QA.....	53
Figure 19. An example of connections set up	56
Figure 20. Data flow of the real-time connections	57
Figure 21. Detailed PQOC/QA algorithm	62
Figure 22. Occupancy distribution analysis for M/G/m under FCFS	68
Figure 23. Connections setup performance	74
Figure 24. Throughput performance of high priority data.	76
Figure 25. Mean delay and jitter performance of VBR traffic	77
Figure 26. Mean delay and jitter performance of VBR traffic	78

Figure 27. Delay bound of VBR traffic.....79

Figure 28. The impact on VBR mean delay under various ABR loads and the ABR delay comparison 80

Figure 29. Mean delay comparison between ABR and VBR traffic under equivalent loads 82

Figure 30. The impact on ABR delay under various burstiness and loads of VBR traffic.....82

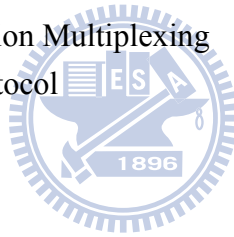


Acronyms

ABR	Available Bit Rate
ADM	Add/Drop Multiplexer
AOTF	Acousto-Optic Tunable Filter
ATMR	Asynchronous Transfer Mode Ring
AWG	Arrayed Waveguide Gratings
BMR	Burst-Mode Receiver
CAC	Call Admission Control
CBR	Constant Bit Rate
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
DAVID	Data And Voice Integration over DWDM
DQDB	Distributed Queue Dual Bus
DQBR	Distributed Queue Bidirectional Ring
DWDM	Dense Wavelength Division Multiplexing
EOTF	Electro-Optic Tunable Filter
FBG	Fiber Bragg Gratings
FCFS	First Come First Serve
FIFO	First In First Out
FPGA	Field Programmable Gate Array
FR	Fixed-Tuned Receiver
FT	Fixed-Tuned Transmitter
FWM	Four Wave Mixing
HORNET	Hybrid Optoelectronic Ring Network
IP	Internet Protocol
MAN	Metropolitan Area Network
M-ATMR	Multiple Asynchronous Transfer Mode Ring
MMPP	Markov Modulated Poisson Process
MMR	Muliple MetaRing
MPU	MAC Processing Unit
MPEG	Moving Picture Experts Group
MTIT	Multitoken Interarrival Time
M/G/m	Multi-Server Queueing System
OCS	Optical Circuit Switching
OPS	Optical Packet Switching
O-Node	Ordinary Node
PDF	Probability Desnsity Function
PQOC	Probabilistic Quota plus Credit



PQOC/QA	Probabilistic Quota plus Credit with QoS Assurance
QoS	Quality of Service
RAM	Random Access Memory
RingO	The Italian Ring Optical Network
RPR	IEEE 802.17 Resilient Packet Ring
SDH	Synchronous Digital Hierarchy
SGDBR	Sampled-Grating Distributed-Bragg-Reflector
SOA	Semiconductor Optical Amplifier
SONET	Synchronous Optical Network
SRR	Synchronous Round Robin
SR ³	Synchronous Round Robin with Reservations
STE	SYNC Timing Extractor
S-Node	Server Node
TDM	Time Division Multiplexing
TR	Tunable Receiver
TT	Tunable Transmitter
VBR	Variable Bit Rate
WDM	Wavelength Division Multiplexing
WDMA	WDM Access Protocol



Chapter 1. Introduction

1.1 Optical WDM Networks

1.1.1 An Overview

For long-haul backbone networks, optical wavelength division multiplexing (WDM) [1,2] has been shown successful in providing virtually unlimited bandwidth to support a large amount of steady traffic based on the optical circuit switching (OCS) paradigm. Future optical metropolitan area networks (MANs) [3,4], on the other hand, are expected to cost-effectively satisfy a wide range of applications having time-varying and high bandwidth demands and stringent delay requirements. Nevertheless, today's metropolitan area networks are mostly SONET/SDH ring networks. These networks are circuit-switched networks. The SONET/SDH technology offers data transmission only at specific rates from a prescribed set of rates. The main drawback of SONET/SDH networks is that due to their time-division multiplex operation in conjunction with a circuit set-up time on the order of several weeks or months [5], they accommodate packet traffic only inefficiently [6], especially when the traffic is highly variable, giving rise to the so called metro gap. Such facts bring about the need of exploiting the optical packet-switching (OPS) [4,7,8] paradigm that takes advantage of statistical multiplexing to efficiently share wavelength channels among multiple users and connections. Note that the OPS technique studied here excludes the use of optical signal processing and optical buffers, which are current technological limitations OPS faces. Numerous topologies and architectures [3,4,7-16] for OPS-based WDM metro networks have been proposed. Of these proposals, the structure of slotted rings [9-16] receives the most attention. Essentially, these slotted-ring networks offer high-performance access and

efficient bandwidth allocation by means of medium access control (MAC) [17-20] schemes.

Regarding the design of the WDM networks, we first consider two of the important issues: node architectures [21,22] and bandwidth reuse [14,16]. In the WDM networks, the nodes are equipped with number of transmitters and receivers to transmit and receive data. The transmitters and receivers are either fixed-tuned to a particular wavelength (denoted as FT/FR) or tunable to any wavelength (denoted as TT/TR). The systems are first designed by a non-scalable architecture which is equipped with the same number of FT/FR as that of the wavelengths [23]. The main advantage of this system is that concurrent transmissions on distinct channels are possible at a given node. While this architecture requires as many wavelengths as there are nodes in the network, and this severely limits the scalability of such a network. Further, the nodes are further designed with advanced optical devices, such as TT-FR and TT-TR structures. Systems based on TT-FR is still incurred a scalability problem, since each node or a group of nodes is assigned a home channel to receive data. Once there is no data to transmit to a particular node, the bandwidth of its home channel is then wasted. Except the throughput degrades due to the static assignment (poor statistic multiplexing gain), the maximum number of nodes is also limited by the number of available channels. While systems based on the TT-TR structure are the most flexible in accommodating a scalable user population but with a most challenging issue in designing and implementing a high-speed photonic hardware component (TR).

We further observe that a ring network with spatial bandwidth reuse achieves much better throughputs than in star topology [24,25], where bandwidth reuse is not possible. Indeed, the advantage of spatial bandwidth reuse is one of the main reasons why the structure of slotted rings receives the most attention. Generally, the spatial

reuse includes source- and destination-stripping schemes. In the case of source-stripping operation, the transmitting node is responsible for marking the slot empty after it has completed an entire ring loop. With destination stripping, the destination node receives the packets and removes them from the ring, making the slot reusable earlier than in the previous scheme. The network capacity of unidirectional ring networks can be increased with destination stripping where multiple simultaneous transmissions can take place on each wavelength. For uniform traffic, the mean distance between source and destination is half the ring circumference. As a consequence, two simultaneous transmissions can take place at each wavelength on average, resulting in a network capacity that is twice as large as that of unidirectional rings with source stripping. However, in this thesis, we propose a new notion which is referred to as server-stripping. Only a few numbers of nodes in the network is capable of removing the data from the ring. The associated network architecture will be shown to be most cost-effective for bandwidth reuse.

1.1.2 Existing MAC Schemes on Single-Channel Rings

Before assessing the OPS WDM slotted-ring networks, we first examine some formerly proposed MAC schemes for ring networks. These schemes can generally be categorized as quota-based or rate-based. In the quota-based schemes, each node is allocated a quota that is the maximum transmission bound within a variable-length cycle. Most of the research work focuses on the dynamic adjustment of the cycle length. In the following, we introduce two of the well-known quota-based schemes: ATMR [3,26] and MetaRing [3,27]. And, we also introduce a rate-based scheme: RPR (IEEE 802.17 Resilient Packet Ring) [28].

The ATMR protocol adopts a quota-based scheme on single/dual- ring network. It provides fairness control with a cycle reset mechanism. The mechanism allocates

each node a maximum transmission bound (quota) within a cycle, and it re-starts a new cycle by sending a reset signal from the last active node. If the last active node detects inactivity of all other nodes, it generates a reset which is sent to all nodes as soon as the node itself stops sending. Monitoring of inactivity is performed as each active node overwrites a busy address field in the header of each cell with its own address. So any node which receives a slot with its own busy address assumes that all other nodes are inactive because none of them has overwritten the field. The reset is responsible for the distributed fairness control and causes a node to set up its window counter to the initial window size. The counter is decremented each time the node fills a free slot with data. By counting it down to zero it is guaranteed that within a reset period, i.e. the time between two consecutive resets, each station uses a maximum number of cells. As the window counter expires, the node is forced into the inactive state. In this state it cannot send any data until the next reset activates once more. If a station has no more data to send the node will pass over to the inactive state, but it may become active again without receiving the next reset on arriving data at the transmit queue. The primary disadvantage of this scheme is that a node cannot send any packet before receiving the reset signal. In other words, there is an idle gap between two consecutive reset periods. Therefore, the bandwidth is waste and system utilization downgrades. Another disadvantage is the determination of the value of quota, which relates to the network throughput and the maximum delay time. Since the reset signal has to run at least one round trip time, the quota can not be set too small causing the maximum delay time is above one round trip time.

MetaRing deploys a quota-based fairness scheme on dual-ring network. This mechanism works with a hardware control message, called SAT-signal. This is very short, and on a dual counter rotating ring it circulates in the opposite direction to the data which it controls. The signal has preemptive resume priority, i.e. at any time it

can be inserted into the data flow. If a station gets the SAT-signal and it is satisfied, it sends the signal immediately to the neighboring node. Otherwise it keeps the signal until it becomes satisfied. A node can transmit its local traffic whenever it has not exhausted its quota. When sending the signal to the neighboring station, the slot counter is reset to zero. That is, the quota of a node is renewed every time SAT-signal visits the node. The major drawback of this global fairness is that quotas can only be renewed when a node receives SAT-signal, and which may need several of ring times depending on the value of quota. Therefore, the maximum access delays are within the order of round trip times. When the ring network is overloaded, the access delays seen by each node will oscillate between zero and the maximum value depending on when a packet comes in relative to the recent SAT-signal visit.

The standard, IEEE 802.17, Resilient Packet Ring (RPR) deploys a rate-based fairness algorithm. Current RPR networks are single-channel systems (i.e., each fiber carries a single wavelength channel) and are expected to be primarily deployed in metro edge and metro core areas. It adopts destination stripping enables nodes in different ring segments to transmit simultaneously, resulting in spatial reuse and increased bandwidth utilization. RPR provides a three-level class-based traffic priority scheme. As a rate-based MAC, an RPR station implements several traffic shapers to smooth and control the rate of each traffic class. The three-level classes: class A (divided into A0, A1) for a low-latency low-jitter class, class B (BCIR, B-EIR) for a class with predictable latency and jitter, and class C be a best effort transport class. The two traffic classes C and B-EIR are called fairness eligible (FE), because such traffic is controlled by a fairness algorithm. The shapers for classes A0, A1, and B-CIR are preconfigured; the bandwidth for class A0 is called reserved. And, the downstream shaper, set to the unreserved rate (other than class A0), ensures that the total transmit traffic from a station does not exceed the unreserved rate. While the FE

shaper is dynamically adjusted by the fairness algorithm for control class B-EIR and class C. RPR also includes a local fairness algorithm to solve the unfairness among the contending stations.

In summary, ATMR allows the last active node to initialize a reset-signal rotating on the ring to inform all nodes to re-start a new cycle. MetaRing uses a token-based signal circulating around the ring. When a node receives the token, it either forwards the token and thus starts a new cycle immediately, or holds the token until the node has no data to send or the quota of previous cycle expires. These schemes were shown to achieve high network utilization and great fairness. However, they cause cycle lengths to prolong several ring times, resulting in a large maximum delay bound and delay jitter, and thus poor bursty-traffic adaptation. In the rate-based schemes, RPR (IEEE 802.17 Resilient Packet Ring) is based on a pre-determined leaky bucket rate to transmit data, in combination with a local-fairness algorithm to resolve the potential congestion problem. Comparing with the quota-based schemes, the rate-based scheme was shown to reduce the maximum delay bound [29]. However, the leaky rate is modified only after receiving the feedback from the downstream nodes when congestion occurs. As a result, due to using the pre-determined rate and the slow response to rate changes, the scheme yields poor statistical multiplexing gain and dissatisfying delay-throughput performance especially under the high-burstiness fluctuating traffic condition. The goal of this thesis is to present a quota-based MAC scheme that tackles the performance problem from a perspective of the determination of the quota rather than the cycle length.

1.1.3 A Survey on WDM Ring Networks

There have been numerous OPS WDM slotted-ring networks proposed in the literature [3]. In the following, we assess three well-known prototyping networks that

are most relevant to our work. First, Hybrid Optoelectronic Ring NETWORK (HORNET) [9] is a bi-directional WDM slotted ring network in which each node is equipped with a tunable transmitter and a fixed-tuned receiver. It employs a MAC protocol, called Distributed Queue Bidirectional Ring (DQBR), which is a modified version of IEEE 802.16 Distributed Queue Dual Bus (DQDB) protocol [30]. DQBR requires each node to maintain a distributed queue via a pair of counters per each wavelength to ensure that packets are sent in the order they arrive at the network. With DQBR, HORNET achieves acceptable utilization and fairness at the expense of high control complexity for maintaining the same number of counter pairs as that of wavelengths. Moreover, due to the use of fixed-tuned receiver, HORNET statically assigns each node a wavelength as the home channel for receiving packets. Such static wavelength assignment results in poor statistical multiplexing gain and thus throughput deterioration.

The second prototyping network, called Ring Optical Network (RingO) [10], which is a unidirectional WDM slotted ring network with N nodes where N is equal to the number of wavelengths. Each node is equipped with an array of fixed-tuned transmitters and one fixed-tuned receiver operating on a given home wavelength that identifies the node. Such a design gives rise to a scalability problem. RingO employs a MAC protocol, called a synchronous round robin with reservations (SR^3) [11], which is a combination of the synchronous round-robin (SRR), token-control quota based (Multi-MetaRing), and slot-reservation mechanisms. The scheme was shown to achieve high utilization and fairness. As for the fairness-scheme, Multi-MetaRing, it inherits all the pros and cons from the MetaRing. Specifically, there are W numbers of tokens rotating on W wavelengths. The scheme encounters an additional problem in which a node may hold several tokens at the same time due to the fact that only one data packet can be sent per slot time. The problem results in an increase in access

delay and throughput degradation.

The metro network of the European IST Data And Voice Integration over DWDM (DAVID) [12,13] attempted to address the overall efficiency of ring-to-ring traffic, and fairness and QoS control inside a metro ring. DAVID is structured to be comprised of several independent fiber rings interconnected via a buffer-less SOA-based packet switch, i.e., the hub node. The hub node is responsible for forwarding data packets among different rings of the network in the optical domain via an available wavelength. Due to having multiple rings, the hub requires each node to make slot reservation prior to the transmissions and has to resolve a feasible wavelength-to-wavelength permutation [15] at all times. Within each ring, the Multi-MetaRing scheme is employed to ensure the fairness control. Each active node is equipped with a tunable transmitters, a tunable receiver, and an SOA-based slot eraser, enabling high slot reuse but at the expense of prohibitive system cost.

Note that both RingO and DAVID adopt Multi-MetaRing as their fairness control scheme. Recall that MetaRing is a quota-based scheme, thereby most relevant to our work. In MetaRing, a control message SAT-signal (which stands for SATisfied) rotates around the ring, and the quota of a node is renewed every time SAT-signal visits the node. In Multi-MetaRing, it is simply designed by independent multiple MetaRing with one separate SAT-signal for each channel (i.e. W number of SAT-signals for W number of data channels). In other words, there are multiple token-like signals rotating on the multiple channels to ensure the fairness among nodes. The quota of a particular channel of a node is renewed only when the node receives the token on that channel. Therefore, Multi-MetaRing inherits all the disadvantages from the MetaRing. That is, the maximum access delays are within the order of round trip times. When the ring network is overloaded, the access delays seen by each node will oscillate between zero and number of round trip times. This

outcome is especially unsuitable for bursty metro traffic and real-time traffic. When applying to WDM networks, Multi-MetaRing encounters an additional problem in which a node may hold several tokens at the same time due to the fact that only one data packet can be sent per slot time (due to the fact that each node is equipped with only one transmitter). The problem results in an increase in access delay and throughput degradation.

1.1.4 Existing MAC Schemes with QoS assurance in WDM Networks

For future optical Wavelength Division Multiplexing (WDM) networks, OPS WDM networks have been envisioned as a future framework for next-generation Internet (NGI), which is expected to support integrated multimedia services with various quality-of-service (QoS) requirements [31-42]. Expected supported services include constant bit rate (CBR), variable bit rate (VBR), and available bit rate (ABR). The real-time traffic, such as CBR and VBR traffic, referred to as high priority data, is subject to a centralized/distributed call admission control (CAC) [43-48] that accepts connections if all demands are guaranteed to be satisfied. While the ABR traffic which is referred to as low priority data takes advantages of all the remaining bandwidth. Pertaining to such OPS WDM networks, one of the most interesting and challenging issues is to design an efficient medium access control (MAC) that flexibly accommodates maximal real-time traffic with remarkable QoS performance while still sustaining exceptional aggregate system throughput.

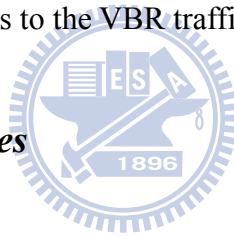
Most existing MAC schemes with QoS provision primarily focus on two major challenges, the reservation mechanism and the accommodation of real-time traffic. In single-channel networks, such as IEEE 802.16 Distributed Queue Dual Bus (DQDB) protocol [30], Asynchronous Transfer Mode (ATM), and IEEE 802.17 Resilient Packet Ring (RPR) [28], support QoS by rate-basis reservation, which allocates the

required rates of bandwidth for real-time traffic. However, in WDM networks, they adopt slot-basis reservation schemes [31-34], where they clearly specify which data slots are reserved for real-time traffic. Since in most of the current WDM networks, each node is equipped with only one receiver, bringing in a receiver-contention problem (two packets destined for the same node are prohibited at the same slot time). If the real-time traffic is only rate-reserved, it may fail to transmit due to the receiver-contention problem. As to regard the accommodation of real-time traffic, most approaches focus on the bandwidth requirements estimation, such as guaranteed bandwidth (peak rate), effective bandwidth, and dynamic measurement bandwidth [43-48]. Both guaranteed bandwidth and effective bandwidth are often over-estimated, thereby resulting in poor system utilization. While the measurement-based bandwidth is too complex and difficult to be properly predicted, it is either over-estimated or under-estimated (poor QoS guarantee). In this thesis, we simply tackle the problem from the perspective of a given proportion of bandwidth left over for the bursty traffic rather than the actual bandwidth estimation.

In WDM ring networks, existing researches propose QoS provision by slot-basis reservation but either in inflexible or over aggressive reserved manner, thereby causing poor statistical multiplexing gain for real-time traffic or system utilization degradation. The methodologies in [31,32] make reservation at their corresponding preferential frame-based slots, which were pre-assigned either on a per-source-destination basis [31] or per-destination basis [32] to suit the hardware limitations imposed by their network architectures. Since each node is equipped with one fixed-tuned receiver tuned to its home channel, the reservation can only be done at some pre-assigned wavelength and at some pre-assigned slot times. While [32] solves the scalability problem, thus the number of the nodes is greater than the number of wavelengths. These schemes indeed satisfy the QoS requirement. However,

because they make reservations only at particular slots, they are rather inflexible and inefficient, leading to poor statistical multiplexing gain for real-time traffic. Another scheme [33] makes high-priority-marks at the control channel and shares with all nodes whenever a node fails to transmit any high priority data. Although the share among nodes confines the total number of reservations and lowers the mean delay, it could still make too many redundant reservations, especially when it deals with highly-bursty traffic (VBR traffic). In such networks, the scheme compensates by compressing the bandwidth for best-effort traffic, thereby degrading the overall system utilization. Despite of the disadvantages discussed above, all the schemes for WDM ring networks focus primarily on the slot-reservation methodology only, lacking an overall evaluation of real-time traffic performance and do not include a viable CAC function which adapts to the VBR traffic.

1.2 Motivation and Objectives



Our major goal has been the design and prototype a high-performance optical packet-switched metro WDM ring network (HOPSMAN). In this thesis, we present the architecture and the MAC scheme of HOPSMAN [7,8]. HOPSMAN has a scalable architecture in which the node number is unconstrained by the wavelength number. Nodes are equipped with high-speed photonic hardware components that are capable of performing nanosecond-order OPS operations. HOPSMAN also encompasses a small number of server nodes that are additionally equipped with optical slot erasers capable of erasing optical slots resulting in an increase in bandwidth efficiency. In essence, HOPSMAN is governed by a novel medium access control (MAC) scheme, called Probabilistic Quota plus Credit (PQOC), which is further enhanced with QoS assurance, called Probabilistic Quota plus Credit with QoS

Assurance (PQOC/QA). The proposed MAC scheme embodies a highly efficient and fair bandwidth allocation in accordance with a quota being exerted probabilistically. Unlike the existing quota-based schemes, our goal is to determine the quota rather than the cycle length. Taking the server-node number and destination-traffic distribution into account, we analytically derive the probabilistic quota. Besides, the MAC scheme introduces a time-controlled credit for regulating a fair use of remaining bandwidth particularly in the metro environment with traffic of high burstiness. Extensive simulation results show that HOPSMAN with our proposed MAC scheme achieves great fairness and exceptional delay-throughput performance under a wide range of traffic loads and burstiness.

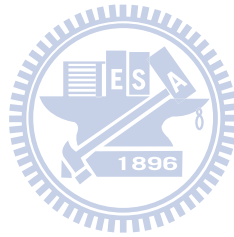
Furthermore, we enhance the MAC scheme with QoS assurance. PQOC/QA not only inherits the original basic design of PQOC, but also integrate with QoS support (it supports CBR, VBR and ABR traffic). To support QoS and to resolve the receiver-contention problem inherent in WDM network, PQOC/QA adopts slot-basis reservation through a simple and flexible marking mechanism, thereby achieving high statistical multiplexing gain for real-time traffic and establishing real-time traffic connections only within a single ring time under normal loads. To adapt to VBR traffic, instead of focusing on estimation of the bandwidth requirements, PQOC/QA employs constant mean rate reservation on each cycle of the ring and well accommodates VBR traffic fluctuation by the remaining quota (excluding the quota used by the reservation). Along with a simple but effective CAC function, the minimum (guaranteed) remaining quota is controlled by a predefined quota ratio. Therefore, if the quota ratio is set reasonably, the probability that the fluctuation of VBR traffic fails to transfer due to expired quota is significantly small. Consequently, PQOC/QA can well accommodate VBR traffic fluctuation, thus achieving exceedingly low VBR delay and jitter. Moreover, through the simple CAC, the

network can simply provide guaranteed load of real-time traffic and also obtain guaranteed setup queueing time. Further, based on a non-aggressive reservation mechanism (mean rate reservation) and a flexible transmission strategy, the overall performance achieves not only QoS assurance, but also retains the maximal system utilization.

Another important goal of our work is to propose the analysis of the expected setup queueing time. The system is modeled as an M/G/m queue under the first-come-first-served (FCFS) service discipline, where the maximum admissible quota of real-time traffic is modeled as the number of servers. Our main contribution is a novel approximation derivation yet accurate results for a multi-server queueing system with the specific service time in our system, an exponential duration plus an extra constant value. Actually, almost no exact results are known for the first moment of the stationary waiting time distribution in a multiple-server queueing system (M/G/m). Exclusively, the accurate results have only been attained for a limited number of special service distributions, such as the exponential [50], deterministic [51], Erlang [52-54] and hyperexponential-2 [54-56] distributions. Most work provides approximation formulas [52-69] which estimated the mean queueing time for the M/G/m queue from the first two or three moments of the service distribution. They propose approximation results only maintained a less than 10% relative error for certain properties of service distributions. In this thesis, we develop a novel approximation for a particular general service distribution, and we show that the analytical results match well with the simulation results of the mean setup queueing time.

1.3 Organization of the Thesis

The remainder of this thesis is organized as follows. In Chapter 2, we present the network and node architectures of HOPSMAN. In Chapter 3, we describe the MAC scheme (PQOC) and delineate the analysis for the determination of the probabilistic quota and also show the simulation results. In Chapter 4, we elaborate on the details of the MAC scheme (PQOC/QA) which enhances PQOC with QoS assurance and present a novel analysis of queueing time by an M/G/m queueing system and also show the simulation results. Chapter 5 focuses on the hardware implementation of the testbed and the demonstration of a potential application for HOPSMAN. Finally, concluding remarks are given in Chapter 6.



Chapter 2. General Network and Node Architectures

2.1 Network Architecture

HOPSMAN is a unidirectional WDM slotted-ring network with multiple WDM data channels (λ_1 - λ_W , at 10 Gb/s) and one control channel (λ_0 , at 2.5 Gb/s), as shown in Figure 1. Channels are further divided into synchronous time slots. Each data-channel slot contains a data packet in addition to some control fields to facilitate synchronization. Within each slot time, all data slots of W channels are fully aligned with the corresponding control slot. Each control slot is then subdivided into W mini-slots to carry the status of W data slots, respectively.

HOPSMAN contains two types of nodes: ordinary-node (O-node), and server-node (S-node). Each node of both types has a fixed transmitter and receiver pair for accessing the control channel. While an O-node is a regular node with only one tunable transmitter and receiver pair for accessing data channels, an S-node is equipped with multiple tunable transmitter and receiver pairs, and an additional

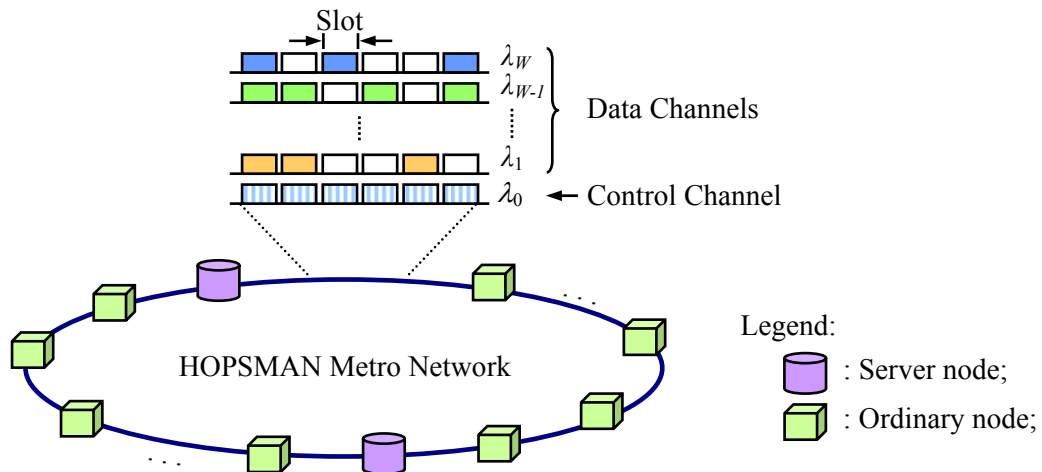


Figure 1. HOPSMAN: network architecture.

optical slot eraser. It is important to note that, HOPSMAN requires at least one S-node, and as will be shown later, bandwidth efficiency improves cost-effectively by using only a small number of S-nodes.

2.2 Node Architecture

The node architecture is shown in Figure 2. It is best described as consisting of two building blocks for control-channel processing and data-channel accessing. For control-channel processing, a fixed optical drop filter (ODF) at the input port first extracts the optical signal from the control channel slot by slot. The control information is electrically received by a fixed-tuned receiver, and processed by the MAC Processor. While the control information is extracted and processed, data packets remain transported optically in a fixed-length fiber delay line. The channel timing processor, in coordination with the SYNC monitoring module, is responsible

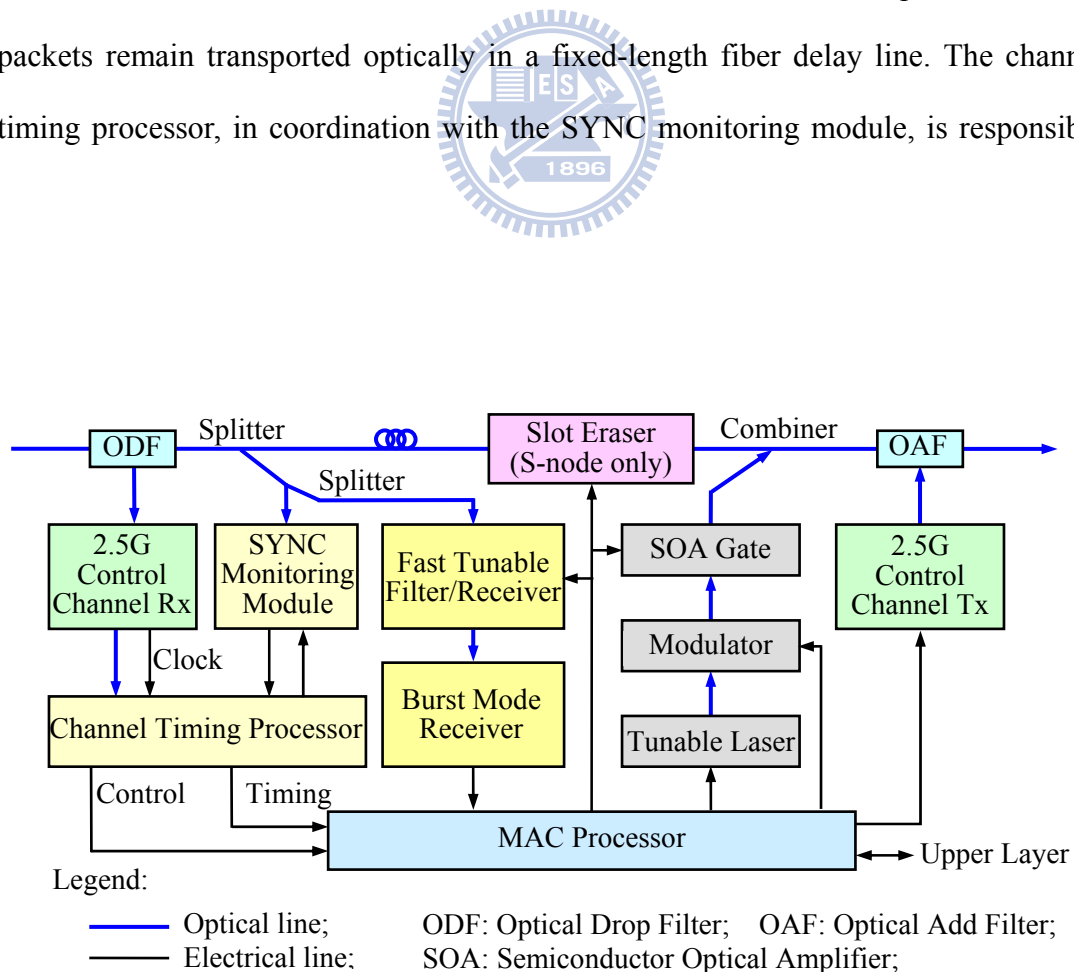


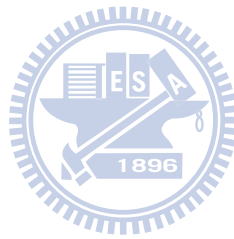
Figure 2. HOPSMAN node architecture.

for extracting the slot boundary timing and for subsequently providing the activation timing for other modules. Having obtained the control information, namely the status of W data channels, the MAC processor then executes the PQOC scheme to determine the add/drop/erase operations for all W channels and the status updates of the corresponding mini-slots in the control channel. Finally, a fixed-tuned transmitter inserts the newly-updated control signal back in the fiber, which is, in turn, combined with data channels' signal via the optical add filter (OAF).

Data-channel accessing corresponds to add and drop operations of data packets based on the broadcast-and-select configuration. Specifically, packets of all wavelengths are first tapped off (“broadcast”) through wideband optical splitters. They are in turn received (“select”) via an optical tunable filter/receiver. The realization and use of such a tunable receiver makes HOPSMAN scalable, namely the number of nodes is no longer constrained by the number of wavelengths. To transmit a packet onto a particular wavelength, the node simply tunes the tunable transmitter to the wavelength. Finally, to discontinue unneeded data packets on any wavelengths, the Slot Eraser (in an S-node only) employs a pair of Mux/Demux and an array of W SOA on/off gates to re-insert new null signals on the wavelengths.

There are three main challenging issues about the hardware implementation of HOPSMAN [7]. They are the synchronization of the data and control channels, and the design and implementation of high-speed photonic tunable receivers, and optical slot erasers. We now briefly describe our solutions to meeting these challenges. First, the channel timing synchronization is ensured via two levels of alignment: coarse-grained and fine-grained synchronization. The first-level coarse-grained synchronization is achieved by inserting a fixed short-fiber delay line in the optical data-channel path to accommodate the basic control computation latency. The fine-grained synchronization is accomplished by matching a fixed-pattern preamble

field (i.e., the SYNC field) at the beginning of each control slot. Second, the fast optical tunable filter/receiver is implemented based on a polarization-insensitive four-wave-mixing (FWM) method, which uses a sampled-grating distributed-Bragg-reflector (SGDBR) fast tunable pumping laser and an SOA. Due to the fact that the receiver's tuning delay solely depends on the tuning speed of the pumping laser, our FWM-based tunable filter/receiver achieves a tuning time of less than 25 ns. Finally, the optical slot eraser has been built with a Mux/Demux pair and an array of SOA gates, which can be turned on/off in 5 ns and achieve an on/off extinction ratio greater than 30 dB.



Chapter 3. MAC Scheme– Probabilistic Quota plus Credit

HOPSMAN is governed by a MAC scheme, called Probabilistic Quota plus Credit (PQOC) [8]. In this section, we first describe the basic concepts of probabilistic quota and credit. We then present the analytic derivation for the determination of the probabilistic quota, which is followed by the detailed algorithm of the scheme.

3.1 Design Principles and the Detailed Algorithm

Before presenting the MAC scheme, we first introduce a term that will be frequently used throughout the rest of the thesis. Since each ordinary node (O-node) has only one tunable receiver, receiver-contention [3] occurs when there is more than one packet destined for the same receiver in a single slot time. Thus, two packets that are destined for the same node are prohibited to simultaneously occupy a single slot time via two different wavelengths. Likewise, because an O-node has only one tunable transmitter, any O-node is restricted to access at most one wavelength in a single slot time. Such a limitation is referred to as the vertical-access constraint.

The entire WDM ring is divided into a number of cycles (see Figure 3), each of which is composed of a pre-determined, fixed number of slots. Basically, PQOC allows each node to transmit a maximum number of packets (slots), or quota, within a cycle. Significantly, even though the total bandwidth is equally allocated to every node by means of the quota, unfairness surprisingly appears when the network load is high. This is because upstream nodes can access empty slots first, resulting in an increasing tendency for downstream nodes to encounter empty slots that are located vertically around the back of the cycle. This issue, as well as the vertical-access constraint, gives rise to poorer throughput and delay performance for downstream

nodes. To resolve the unfairness problem, the quota is exerted in a probabilistic rather than a deterministic fashion, as “probabilistic quota” implies. In other words, rather than transmitting packets immediately if there remains quota (and idle slots of course), each node makes the transmission decision according to a probability. For example, the probability is set to be equal to the quota divided by the cycle length. The determination of the probabilistic quota will be detailed in the following subsection. Such an approach evenly distributes the idle slots within the entire cycle at all times and thus eliminates unfairness against downstream nodes. It is worth noting that, using the probabilistic quota a node may end up making fewer packet transmissions than its quota. This is caused by failing to find idle slots when the access is permitted according to the probability. The problem can be simply resolved by unconditionally granting a packet transmission in a subsequent slot time when there exists an idle slot.

Furthermore, if a node cannot use up its entire quota in a cycle, i.e., has fewer packets than its quota, the node yields the unused bandwidth (slots) to downstream nodes. In return, the node earns the same number of slots as credits. These credits allow the node to transmit more packets beyond its original quota in a limited number of upcoming cycles, called the window. That is, the credits are only valid when the number of elapsed cycles does not exceed the window. The rationale behind this design is to regulate a fair use of unused remaining bandwidth particularly in the metro environment with traffic that is bursty in nature. Notice that there are system tradeoffs in PQOC involving cycle length and window size. For example, the smaller the cycle length, the better the bandwidth sharing; the larger the window size, the better the bursty-traffic adaptation, both at the cost of more frequent computation. The cycle length and window size can be dynamically adjusted in accordance with the monitored traffic load and burstiness via network management protocols. These issues go beyond the scope of this thesis.

The implementation of PQOC is fairly simple. As shown in Figure 3, each control slot contains a header (for synchronization purpose), and W mini-slots carrying the statuses of the corresponding W data slots. There are four distinct states for each data slot- BUSY, BUSY/READ (BREAD), IDLE, and IDLE/MRKD (IMRKD). A node wishing to transmit in a cycle and attaining access permission on the basis of probabilistic quota must first find an IDLE slot. If it succeeds, the node transmits the packet and alters the state from IDLE to BUSY. Otherwise, the node unconditionally transmits its packet on the next available slot without casting the probability again. A destination node that has successfully dropped a packet modifies the slot state from BUSY to BREAD. This allows the next S-node to erase the data slot by changing the status from BREAD back to IDLE, enabling slot reuse by downstream nodes. Furthermore, if a node has no packets to transmit but attains access permission on the basis of probabilistic quota, the node then earns the remaining number of slots as credits for future use, by altering the same number of data slots from IDLE to IMRKD. Finally, a node uses its credits to transmit more packets beyond the probabilistic quota on any IMRKD data slots within the window, and subsequently updates the state to BUSY.

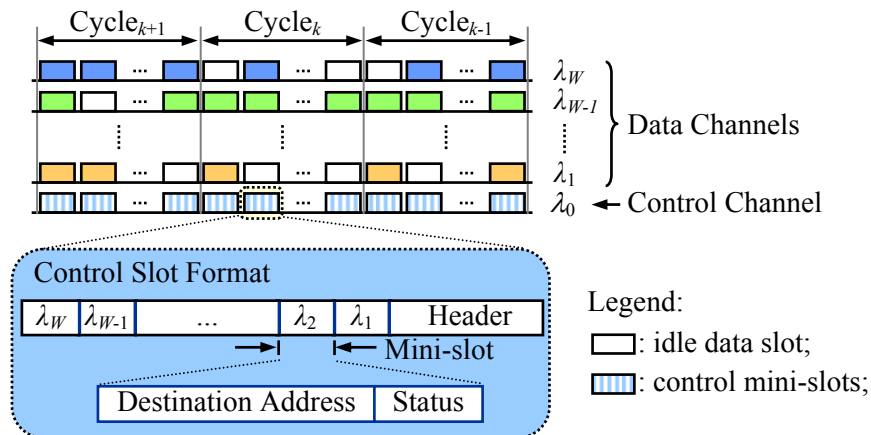
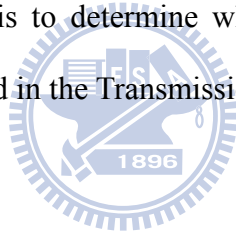


Figure 3. Cycle and slot structures.

The detailed PQOC algorithm is given in Figure 4. An np_{queue} number of packets that have newly arrived or failed to transmit in the previous cycle are scheduled to transmit in the current cycle. The algorithm is executed on a per-slot basis. If the slot is marked as the beginning of a cycle, the algorithm determines three variables: P_Q , the number of idle slots to be marked (ns_{imrkd}), and the number of credits available up to this cycle (nc). Basically, P_Q is computed according to Equation (6) which will be detailed in the next subsection; ns_{imrkd} takes the greater value between 0 and $(Q - np_{queue})$, and nc is calculated on a sliding window basis. nc can also be given based on a fixed window rather than the sliding window strategy. Usually, a sliding window strategy is superior to a fixed window strategy with respect to bursty-traffic adaptation, albeit at the cost of requiring greater computing effort and storage needs. Finally, the primary work of the algorithm is to determine whether transmission of packets is allowed or not, as clearly depicted in the Transmission process (Step 5-6) in Figure 4.



Variables

Q : quota;
 nc : number of credits to be used;
 ns_{imrkd} : number of idle slots to be marked;
 np_{PQ} : number of packets allowed to be transmitted based on prob. quota;
 np_{queue} : number of packets in the queue;
 ws : credit window size;
 $q[i]$: number of remaining quota in cycle i ;
 $c[i]$: number of credits used in cycle i ;

Slot type

Header : {CYCLE_BEGIN, NORMAL};
Status : {BUSY, BREAD, IDLE, IMRKD};

Main Process() /*execute at each slot time*/

1. Read the control slot;

/* Computation at the cycle begin */

2. **if** (slot's *Header* is CYCLE_BEGIN)

/*enter the m^{th} cycle*/

Add the number of arrivals of pre-cycle to np_{queue} ;

Determine P_Q according to Equation (6);

$ns_{imrkd} = \max(0, Q - np_{queue}); np_{PQ} = 0;$

$nc = \max(0, \min(np_{queue} - Q, \sum_{i=m-ws}^{m-1} (q[i] - c[i])))$;

$q[m] = Q; c[m] = 0;$

endif

/* Receive packets */

3. Receive the packet destined to it,
and update BUSY to BREAD;

/* Server node erases packets */

4. **if** (node is S-node)

Erase BREAD packets;

Update BREAD to IDLE;

endif

Figure 4. Detailed PQOC algorithm.

```

/* Transmission process */
5. if (Randomize( $P_Q$ ))
     $np_{PQ}++$ ; /*store transmission allowance */
    endif
6. if (exist a vertical-access-constraint-free packet)
    switch
        case (  $nc > 0$  and exist(IMRKD) ):
            /*transmit by credit */
             $nc --$ ;  $c[m] ++$ ;
            Transmit the packet;  $np_{queue} --$ ;
            Update IMRKD to BUSY;
            break;
        case (  $np_{PQ} > 0$  and exist(IDLE) ):
            /*transmit by quota using IDLE */
             $np_{PQ} --$ ;  $q[m] --$ ;
            Transmit the packet;  $np_{queue} --$ ;
            Update IDLE to BUSY;
            break;
        case (  $np_{PQ} > 0$  and exist(IMRKD) ):
            /*transmit by quota using IMRKD */
             $np_{PQ} --$ ;  $ns_{imrkd}++$ ;  $q[m] --$ ;
            Transmit the packet;  $np_{queue} --$ ;
            Update IMRKD to BUSY;
            break;
        otherwise
            No transmission;
    endswitch
endif
/* Mark slots */
7. if ( $ns_{imrkd} > 0$  and exist(IDLE))
     $ns_{imrkd} --$ ;
    Update IDLE to IMRKD;
endif

```

Figure 4. Detailed PQOC algorithm (continue).

3.2 Bandwidth Allocation- Probabilistic Quota Determination

Assume that there are S server-nodes (S-node 1 to S-node S) in the network dividing itself into S sections (sections 1 to S). Each section contains more than one node including the server node of the section. To simplify the illustration, the S-node for a section is placed in the most downstream location in that section. Namely, section 1 is preceded by S-node S in section S ; and section k is preceded by S-node $k-1$ in section $k-1$, for $k=2$ to S . More specifically, for a network with N nodes, we have $N = \sum_{k=1}^S N_k$, where N_k is the total number of nodes in section k .

Moreover, a slot passed by an S-node is considered as either *Available Bandwidth (AB)* if the slot is empty or erased, or *used bandwidth (UB)* if the slot is non-empty and cannot be erased (have not been read) (see Figure 5). Thus, the quota for a node can be computed as the mean value of the total amount of AB observed by a section divided by the total number of nodes in the section. For instance, in an observed section (referred to as section b), we attain $Q_b = \overline{AB}_b / N_b$, where Q_b denotes the quota to be allocated to any node in section b , \overline{AB}_b the mean value of the total amount of AB passed down by S-node $b-1$, and N_b is the total number of nodes in section b . Notice that the value of \overline{AB}_b is relevant to the traffic destination distribution; and S-nodes are possible to receive more traffic than O-nodes. Accordingly, we derive in the sequel a closed form for \overline{AB}_b under two different destination distributions.

Case 1

In this case, traffic is uniformly distributed to all nodes. Moreover, for simplicity we consider a prevailing case in which S-nodes are evenly located in the network, namely $N_k = N/S$, where $k=1$ to S . Accordingly, each node is given the same \overline{AB} and quota Q , where $Q = \overline{AB}/(N/S)$, for all nodes in the network.

The value of \overline{AB} can be computed as the mean total bandwidth (total number of slots in a cycle) minus \overline{UB} . Thus, in the sequel we analyze the \overline{UB} value passed through S-node $b-1$. The analysis is presented in two parts: one is to consider the transmissions from any section to itself, and the other is to consider the transmissions from a section to the other sections. For the first part, since the total amount of traffic (slots) generated from any section (take section k as an example) is $Q \cdot (N/S)$, thus the traffic amount from section k to section k itself is $Q \cdot (N/S)/S$. Within this amount of traffic, the proportion $Q \cdot (N/S)/2S$ will be erased by the most downstream node (S-node) of the section. Notice that erasable traffic corresponds to the traffic sent from upstream to downstream nodes within this section. Therefore, the remaining traffic, $Q \cdot (N/S)/2S$, which is sent from downstream to upstream nodes within this section, will

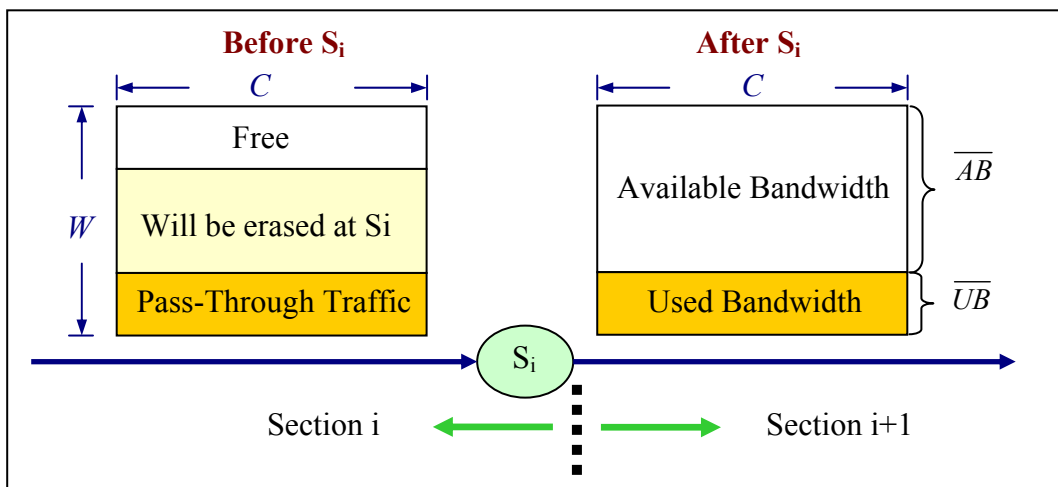


Figure 5. Quota Determination.

be passed through the entire ring and seen by the section as UB .

For the second part, take section $b+2$ as an example. The traffic that is sent from section $b+2$ and passed through the entire ring and seen by S-node $b-1$ and section b as UB is the sum of the traffic destined to sections b and $b+1$. Thus, the UB is equal to $2 \cdot Q \cdot (N/S)/S$. Finally, with all sections taken into account by summing all UB which passes through S-node $b-1$, and given that the total bandwidth in a cycle is $C \cdot W$, where C is the total number of slots in a cycle and W the number of wavelength channels, we obtain \overline{AB} as the following equation

$$\overline{AB} = C \cdot W - \sum_{k=1}^S \left\{ Q \cdot \frac{N}{S} \left(\frac{1}{2S} + \frac{k-1}{S} \right) \right\}. \quad (1)$$

With the equation, $Q = \overline{AB}/(N/S)$, we can attain the closed form solution for Q ,

as

$$Q = \frac{C \cdot W}{N} \left(\frac{2S}{S+2} \right). \quad (2)$$



Case 2

In this case, S-nodes are to receive additional traffic amount than O-nodes. We derive a closed form for \overline{AB}_b under an assumption that a probability p_A of destination traffic is uniformly distributed among all nodes (including the S-nodes), while the remaining probability $1-p_A$ ($=p_S$) of traffic is additionally destined to all S-nodes. Clearly in the case of $p_S=0$, destination traffic is uniformly distributed among all nodes in the network. Regarding the value of p_A , it can be obtained through periodic traffic monitoring via network management protocols, which are beyond the scope of this thesis.

Similar to case 1, to compute the value of \overline{AB}_b , we analyze the \overline{UB} value passed

through S-node $b-1$. The value of \overline{AB}_b can be computed as the mean total bandwidth (total number of slots in a cycle) minus the mean UB . Thus, in the following we analyze the mean UB passed through S-node $b-1$. The analysis is presented in two parts: one is to consider the transmissions from any section to itself, and the other is to consider the transmissions from a section to the other sections. For the first part, since the total amount of traffic (slots) generated from any section (take section k as an example) is $Q_k \cdot N_k$, thus the traffic amount from section k to section k itself is $(Q_k \cdot N_k \cdot p_S / S) + (Q_k \cdot N_k \cdot p_A \cdot N_k / N)$. Of this amount of traffic, the fraction $(Q_k \cdot N_k \cdot p_S / S) + (Q_k \cdot N_k \cdot p_A \cdot N_k / 2N)$, will be erased by section k 's most downstream node, i.e., S-node k . Notice that the second term corresponds to the traffic sent from upstream to downstream nodes within section k . Therefore, the remaining traffic, $Q_k \cdot N_k \cdot p_A \cdot N_k / 2N$, which is sent from downstream to upstream nodes within section k , will be passed through the entire ring and seen by section b as UB .

For the second part, let's take section $b+2$ as an example. The traffic that is sent from section $b+2$, passed through the entire ring, and seen by S-node $b-1$ and section b as UB , is the total amount of traffic destined to sections b and $b+1$. Thus, the mean UB becomes $Q_{b+2} \cdot N_{b+2} \cdot ((p_S / S) + (p_A \cdot N_b / N)) + Q_{b+2} \cdot N_{b+2} \cdot ((p_S / S) + (p_A \cdot N_{b+1} / N))$. Finally, with all sections taken into account by summing all UB which pass through S-node $b-1$, and given that the total bandwidth in a cycle is $C \cdot W$ (where C is the total number of slots in a cycle and W the number of data channels), we obtain \overline{AB}_b as the following equation

$$\overline{AB}_b = C \cdot W - \sum_{k=1}^S \left\{ Q_k N_k \left(\frac{p_A N_k}{2N} + U_k \right) \right\}, \text{ where}$$

$$U_k = \begin{cases} \sum_{m=b}^{k-1} \left(\frac{p_s}{S} + \frac{p_A N_m}{N} \right) & , \text{if } b \leq k \leq S \\ \sum_{m=b}^S \left(\frac{p_s}{S} + \frac{p_A N_m}{N} \right) + \sum_{n=1}^{k-1} \left(\frac{p_s}{S} + \frac{p_A N_n}{N} \right) & , \text{if } 1 \leq k < b \end{cases} \quad (3)$$

Notice that Equation (3) cannot be solved because there is more than one unknown variable (\overline{AB}_b and Q_k 's) in the equation. In the following, we solve the equation under a prevailing case in which S-nodes are evenly located in the network, namely $N_1=N_2=N_3=\dots=N_S=N/S$. In this case, due to the same behavior of S-nodes, we obtain the same quota Q , where $Q = \overline{AB}/(N/S)$, for all nodes in the network. With this additional equation and the simplified version of Equation (3) as

$$\overline{AB} = C \cdot W - \sum_{k=1}^S \left\{ Q \cdot \frac{N}{S} \cdot \left(\frac{p_A}{2S} + \frac{k-1}{S} \right) \right\}. \quad (4)$$

we can attain the closed form solution for Q , as

$$Q = \frac{C \cdot W}{N} \left(\frac{2S}{S - p_s + 2} \right). \quad (5)$$

With the quota determined, we are now ready to obtain the probabilistic quota, denoted as P_Q . Given the total number of packets currently in the queue as np_{queue} , P_Q can simply be expressed as

$$P_Q = \min \left(\frac{Q}{C}, \frac{np_{queue}}{C} \right). \quad (6)$$

3.3 Simulation Results

In this section, we present simulation results to demonstrate the performance of HOPSMAN with respect to throughput, access delay, and fairness. The settings of parameters for simulation are given in the following. The network has a total of 20 nodes ($N=20$), in which node 1 is always designated as an S-node. There are 20 cycles on the ring. Each cycle consists of 100 slots per wavelength. Without specific indication, traffic destinations are assumed to be uniformly distributed among all nodes ($p_S=0$). The credit window size is 10. Traffic is generated following either a Poisson distribution or a two-state (H and L) Markov Modulated Poisson Process (MMPP) [70] for modeling smooth and bursty traffic, respectively. Specifically, the MMPP is characterized by four parameters (α , β , λ_H , and λ_L), where α (β) is the probability of changing from state H (L) to L (H) in a slot, and λ_H (λ_L) represents the probability of arrivals at state H (L). Accordingly, given $\lambda_L=0$, the mean arrival rate can be expressed as $\beta \times \lambda_H / (\alpha + \beta)$, and traffic burstiness (B) can be given by $B = (\alpha + \beta) / \beta$. Finally, simulation is terminated after reaching a 95% confidence interval. Due to having multiple S-nodes, the *traffic intensity* (TI) to be generated per slot per wavelength is unequal to the normalized load (L) per slot per wavelength. They can be related, however, according to Equation (5), as

$$TI = L \cdot \frac{Q}{C \cdot W / N} = L \cdot \left(\frac{2S}{S - p_S + 2} \right). \quad (7)$$

From Equation (7), given S S-nodes in the network, the maximum value of TI (defined as the maximum throughput (T_{max}^S)) occurs at the normalized load L being equal to one. That is,

$$T_{max}^S = \frac{2S}{S - p_S + 2}. \quad (8)$$

From Equation (8), we observe that T_{max}^S increases when p_S is increased. This is because more traffic destined to S-nodes will enable more data slots being erased, i.e., more usable bandwidth after S-nodes. Consequently, if the network inspects that an extra percentage of traffic is additionally and constantly being sent to S-nodes (p_S), it can re-compute and extend the value of quota for each user achieving greater throughput. As previously described, the determination of the p_S value is a network management issue and out of the scope of the thesis.

We further compare the bandwidth efficiency (η) among the three different bandwidth reuse strategies: source stripping, destination stripping and server stripping. The server stripping indicates that the data can only be removed from the S-nodes and the bandwidth can be reused by the downstream nodes. Assume that the bandwidth efficiency of source stripping is 1 ($\eta = 1$). As to destination stripping, the network capacity is almost twice as large as that of unidirectional rings with source stripping under uniform traffic ($\eta \cong 2$). Regarding the server stripping, based on Equation (8) and with $p_S=0$, we obtain $\eta = \frac{2S}{S+2}$. Consequently, we draw the results of HOPSMAN bandwidth efficiency as shown in Figure 6. The figure shows that the bandwidth efficiency rises most dramatically when the network is equipped with only a few number of server nodes. However, as the number of server nodes increases, the increase of the network capacity is quite limited. Note that the destination stripping is equivalent to that the system is with each node as a server node. The result shows that the server stripping is more cost-effective than the destination stripping; therefore, the server stripping is considered as an excellent bandwidth reuse scheme.

In Figure 7, we first draw comparisons of the analytic and simulation results on system throughput under different S-node numbers and two different tunable-transceiver-pair settings. Analytic results are obtained based on Equation (7)

and (8). In the simulation, we assume there are 60 nodes in the network. The results plotted in Figure 7(a) and (b) are obtained from the system with one and two pairs of optical tunable transceivers at each node, respectively. First, we observe from Figure 7(a) that due to the vertical-access constraint when each node is equipped with only one pair of tunable transceivers, the resulted throughput performance is lower than the theoretical maximal throughput that is derived by the above analysis. To justify this fact and to demonstrate the validity of the analysis, we use the second setting in which each node is equipped with two pairs of tunable transceivers, with the result that the vertical-access constraint is lifted. Results are shown in Figure 7(b). From the figure, analytic results are shown to be in profound agreement with the simulation results. Moreover, it is clear that increasing the S-node number results in an improvement in throughput, but at a declining rate as the number of S-nodes grows. For example, the throughput improves most dramatically when the network changes from having one S-node to two S-nodes. The result explains the economy and efficiency of

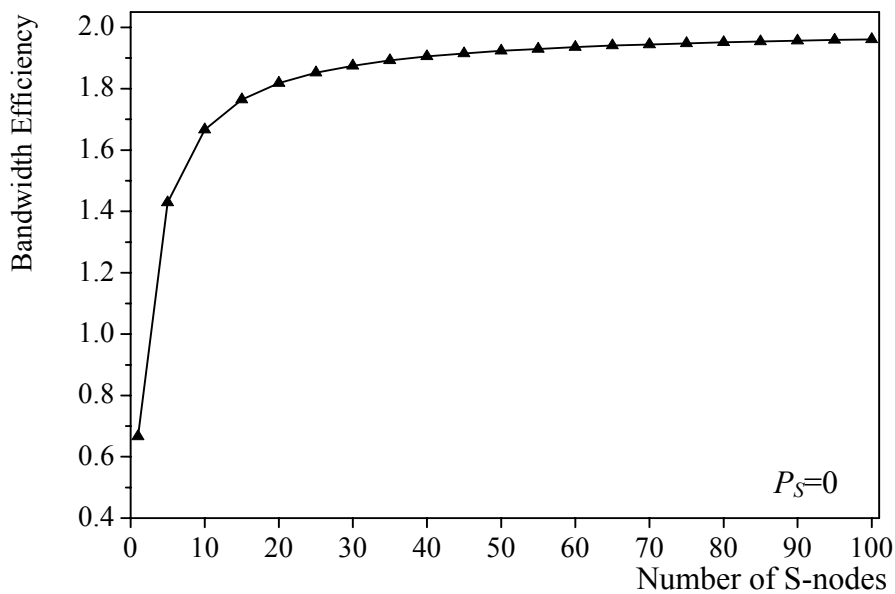
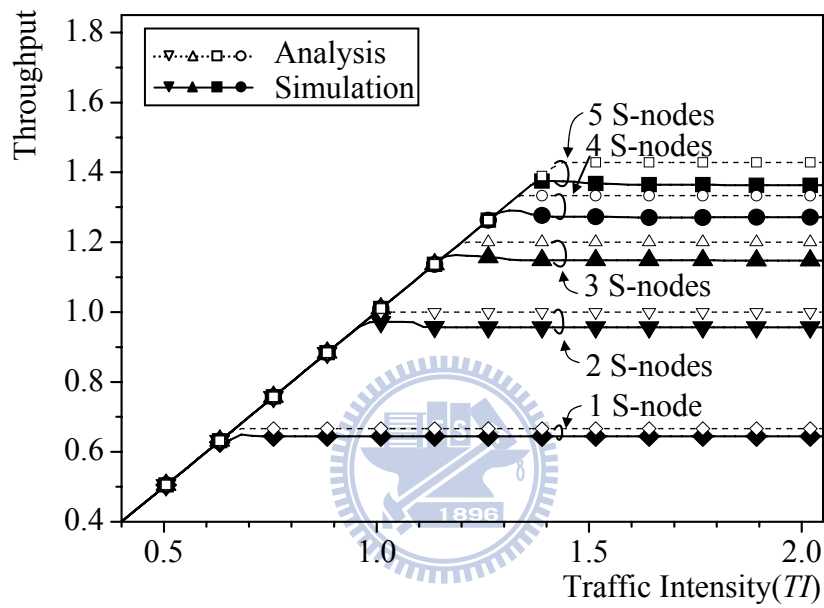


Figure 6. Bandwidth efficiency of HOPSMAN.

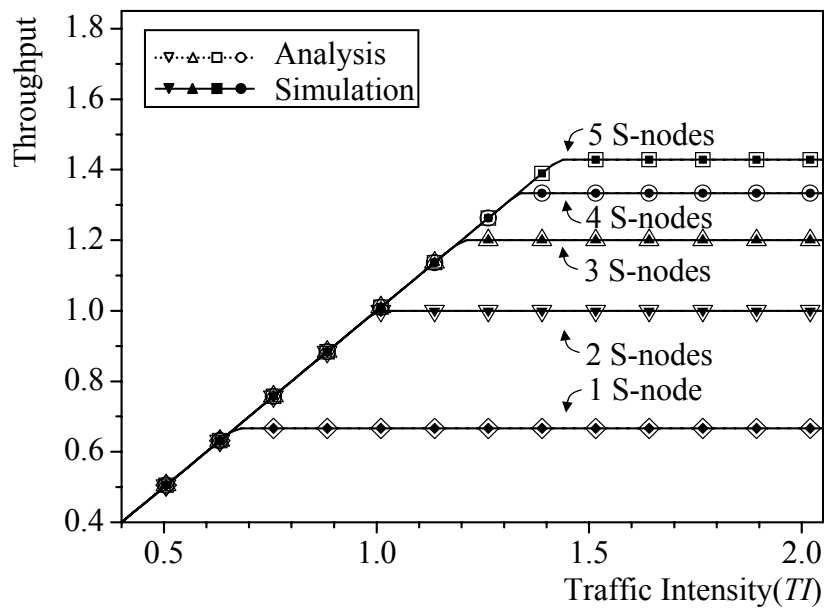
HOPSMAN behind the scarce use of S-nodes.

Focusing on the network with one S-node (Node 1), we next examine the throughput and delay performance of HOPSMAN under various loads and burstiness.

Simulation results are displayed in Figure 8 and 9. As depicted in Figure 8, despite the



(a) Throughput comparison (one TT-TR pair)



(b) Throughput comparison (two TT-TR pairs)

Figure 7. Analytic and simulation results on system throughput under different S-node numbers.

vertical-access constraint, the probabilistic-quota design helps HOPSMAN achieve 100% throughput and fairness under all loads that are less than or equal to 0.9. However, as the network becomes highly saturated when the load reaches 0.99, we observe inevitable throughput deterioration for downstream nodes as a result of the vertical-access constraint. An intensive comparison of delay with and without the probabilistic-quota design will be given shortly. We show in Figure 9(a)-(c) that HOPSMAN guarantees delay fairness under all non-saturated loads. As expected, delay increases with the traffic burstiness. Most importantly, as shown in Figure 9(a), HOPSMAN achieves remarkably low delay under $L=0.7$. Taking this result, along with other results in Figure 8 taken into consideration, it is clear that HOPSMAN achieves superior bandwidth allocation under heavier loads while being able to provide random access under lighter loads.

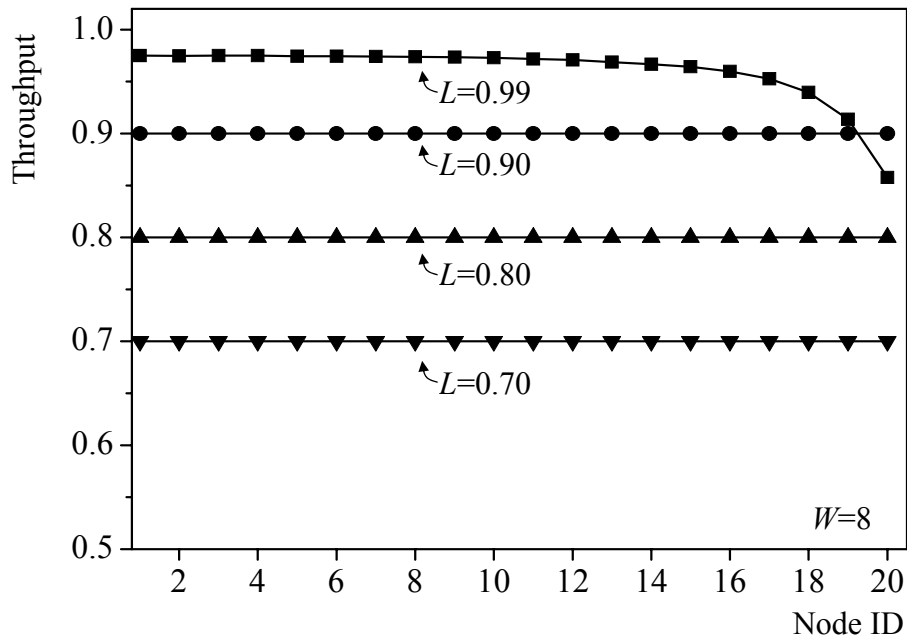
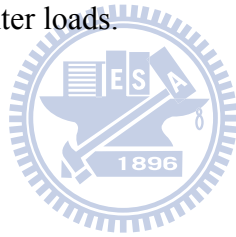
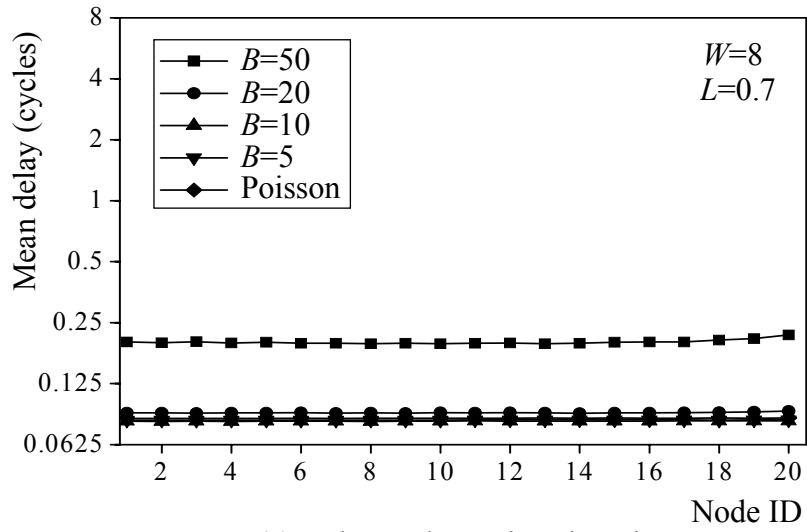
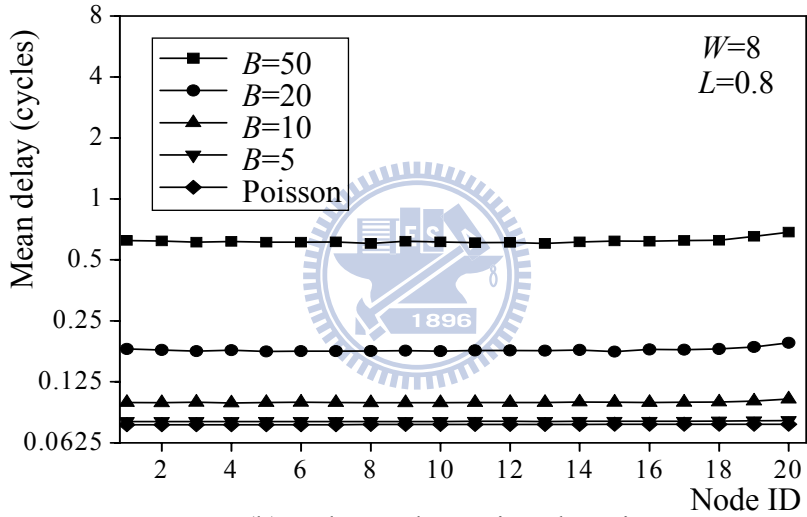


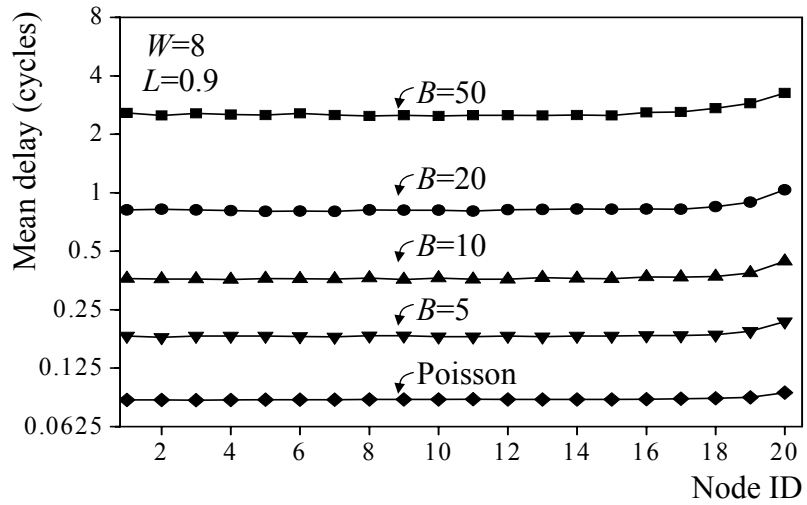
Figure 8. Throughput performance of HOPSMAN.



(a) Delay under various burstiness



(b) Delay under various burstiness



(c) Delay under various burstiness

Figure 9. Delay performance of HOPSMAN.

We now study the impact of credit window size on mean access delay under various traffic burstiness. Simulation results are plotted in Figure 10. As shown in the figure, mean delay declines with increasing window size under all burstiness. However, as was mentioned previously, larger window sizes result in higher computational complexity. Fortunately, we have discovered that such delay reduction occurs most effectively around smaller window sizes (less than ten). Thus, the results can serve as a guideline on the determination of an appropriate and small window size satisfying an acceptable grade of delay.

Recall that in PQOC one uses credit to access slots beyond the quota in order to exert fair access control of remaining unused bandwidth on all nodes. This is demonstrated in the following via simulation for a network with malicious nodes. In the simulation, nodes 5 and 15 are malicious nodes. While each malicious node generates excessive loads of 0.09, each other node generates a load of 0.045, rendering the network highly saturated, namely with a total load of 0.99. Simulation results are displayed in Figure 11. In the figure, we draw a comparison of delay between the PQOC scheme (with credit) and PQC (without the credit) under two burstiness of traffic. As shown in Figure 11(a), under low-burstiness traffic, PQOC makes the two malicious nodes suffer severe delay while leaving other normal nodes completely unaffected. On the contrary, the PQC scheme without credit gives rise to delay deterioration (and thus unfairness) to the neighboring nodes of the two malicious nodes. As the traffic burstiness increases, the delay unfairness problem worsens as shown in Figure 11(b). In this case, PQOC can still guarantee a high grade of fairness among all nodes, except for several downstream nodes due to network saturation. Thus, the PQOC scheme is evidently robust and fair even when under attack by malevolent nodes.

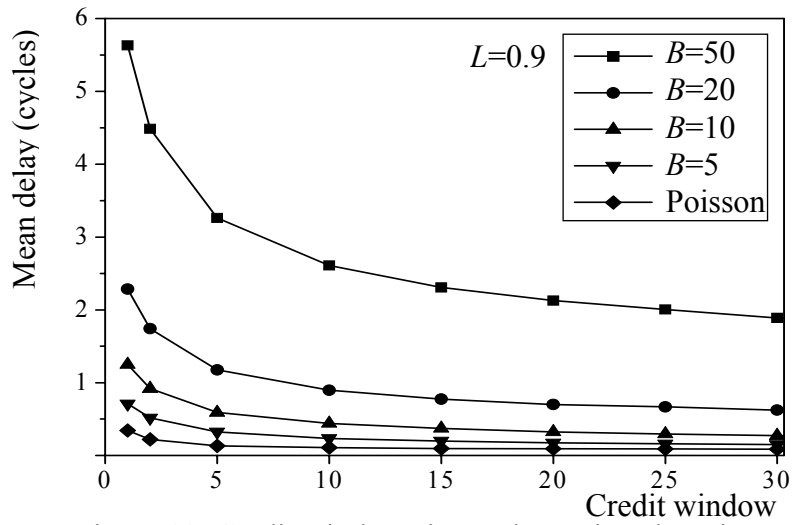
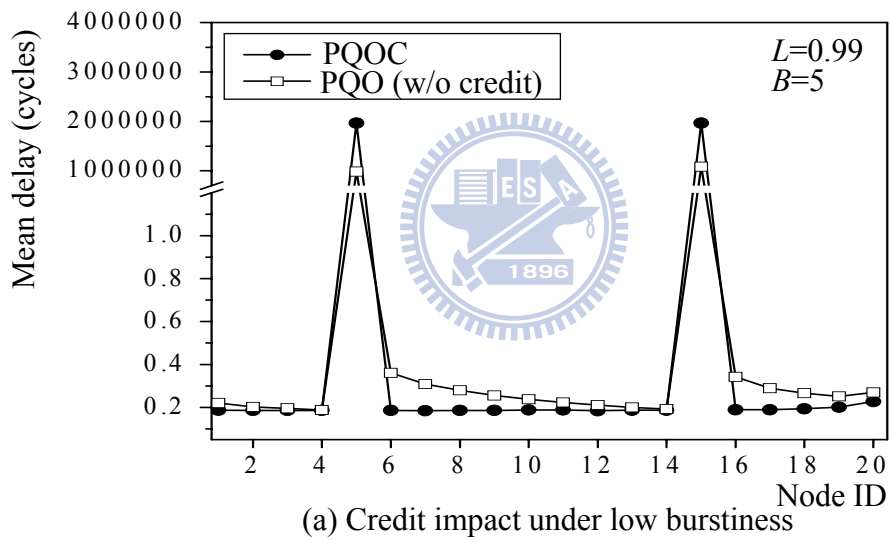
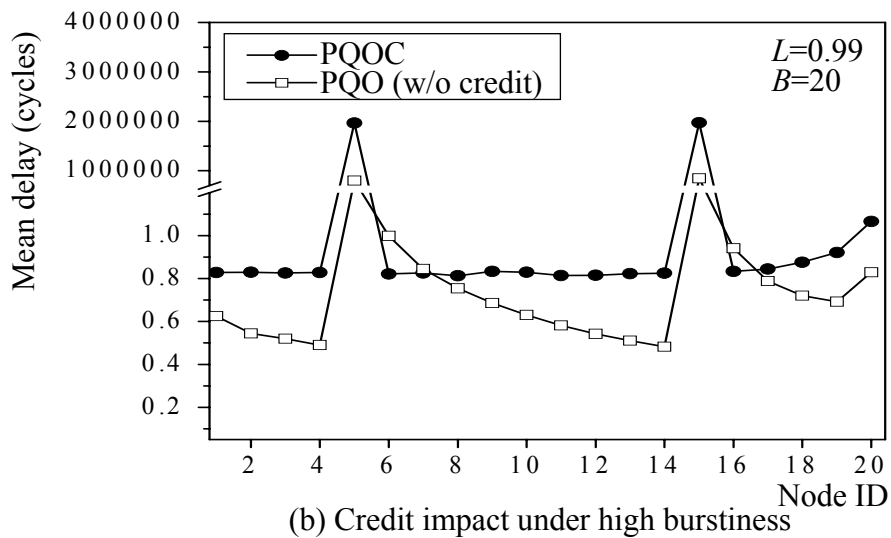


Figure 10. Credit window size under various burstiness.



(a) Credit impact under low burstiness



(b) Credit impact under high burstiness

Figure 11. Credit impact on delay for network with malicious nodes (nodes 5 and 15).

We next examine in Figure 12 the impact of the probabilistic quota design on access delay under a variety of loads and burstiness. In the simulation, we draw a delay comparison between the PQOC scheme and the QOC scheme where the quota is exerted in a deterministic manner. As shown in Figure 12(a), under medium loads of traffic, both schemes yield superlatively low delay. However, as the burstiness increases, despite the fact that QOC attains slightly better delay due to deterministic transmissions, the scheme begins showing signs of delay unfairness for downstream nodes, as shown in Figure 12(b). Worst of all, as the load increases (see Figure 12(c) and (d)), as a result of the vertical-access constraint, QOC scheme not only undergoes

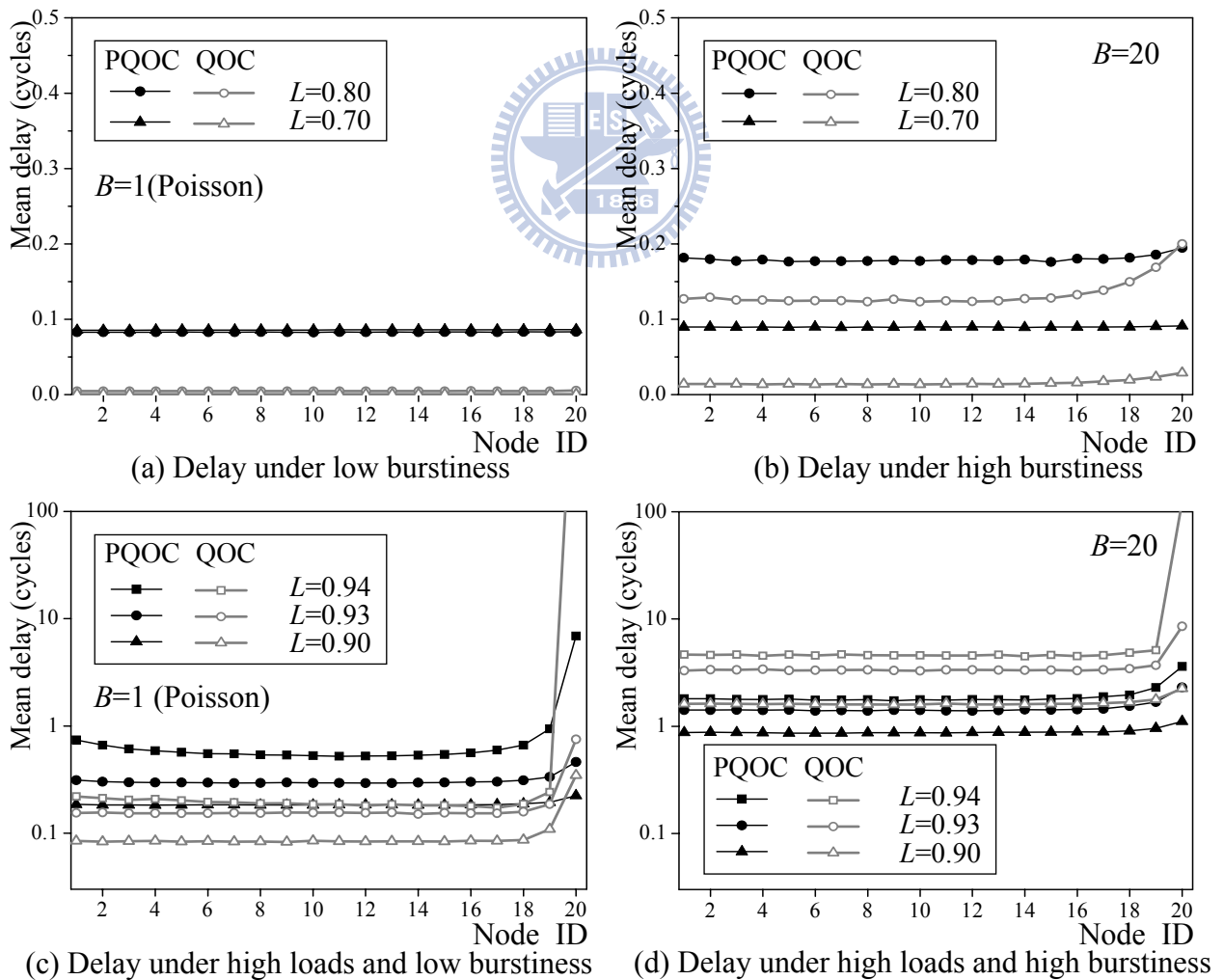


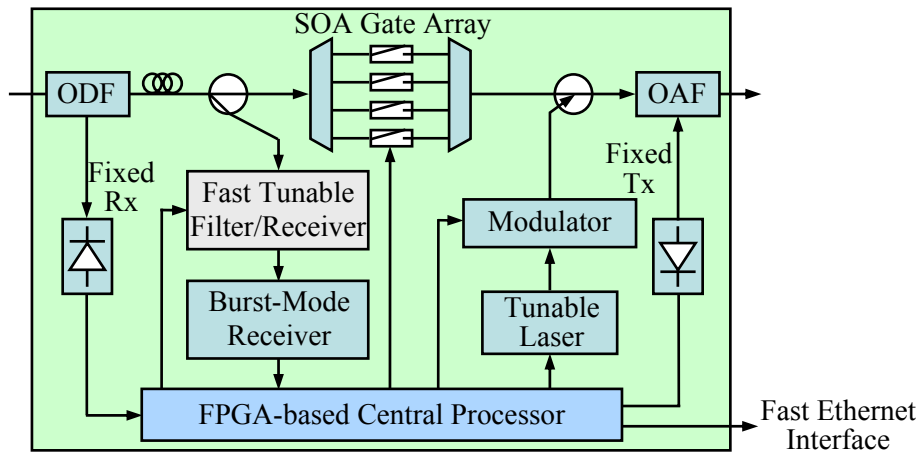
Figure 12. The impact of probabilistic exertion of quota under various loads and burstiness.

severe unfairness toward downstream nodes but also incurs deteriorating delay for all nodes. In other words, PQOC is superior to QOC at a negligible cost. (It is worth noting that PQOC still incurs minor unfairness under exceedingly high loads due to the vertical-access constraint. Such problem can be mitigated by applying un-equal probabilistic quotas to nodes of different locations.) Consequently, the PQOC scheme invariably achieves superior delay and fairness irrespective of traffic loads and burstiness.

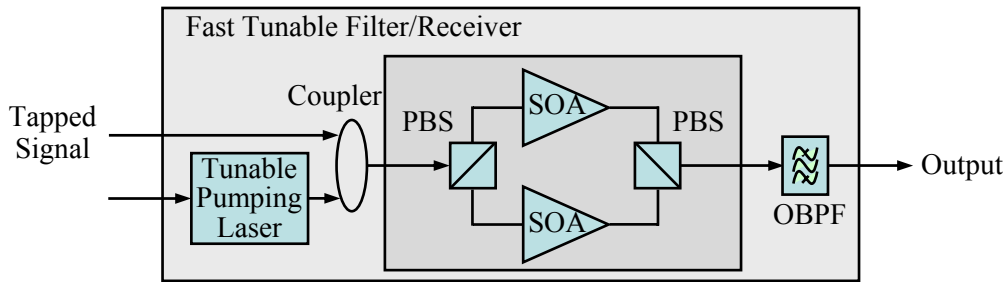


3.4 Testbed Implementation and Experimental Results

We built an experimental ring testbed to demonstrate the feasibility and performance of HOPSMAN [7]. The testbed consists of three nodes: one S-node and two O-nodes (O₁-node and O₂-node). The hardware implementation of an S-node is illustrated by the functional diagrams in Figure 13. Note that the implementation for an O-node is the same as that of an S-node except with the slot eraser removed. The ring testbed is 38.3 km long, with 10 cycles per ring, 50 slots per cycle, and each slot 320 ns long, yielding a total of 500 time slots, or 160 μ s in one ring length. The testbed uses a control-channel wavelength of 1540.56 nm, and four data channels at wavelengths of 1551.72 nm, 1553.33 nm, 1554.94 nm and 1556.55 nm. The input and output power per channel is kept at -10 dBm and 0 dBm, respectively, by using attenuators and amplifiers on the ring. The control channel employs continuous-mode transmission at a rate of 2.5 Gb/s, and is processed at each node through an O-E-O conversion. On the other hand, data channels adopt burst-mode transmissions at a target rate of 10 Gb/s. Owing to the technological immaturity for high-speed optical burst-mode receivers (BMRs), we have deliberately downgraded the data channels' bit rate to 1.25 Gb/s so that commercially available BMRs could be used. It is important to note that the HOPSMAN testbed has been designed so that the rates of the data and control channels are independent of each other. Because of the extensive use of BMRs in passive optical networks, we expect that 10-Gb/s BMRs will be commercially available soon.

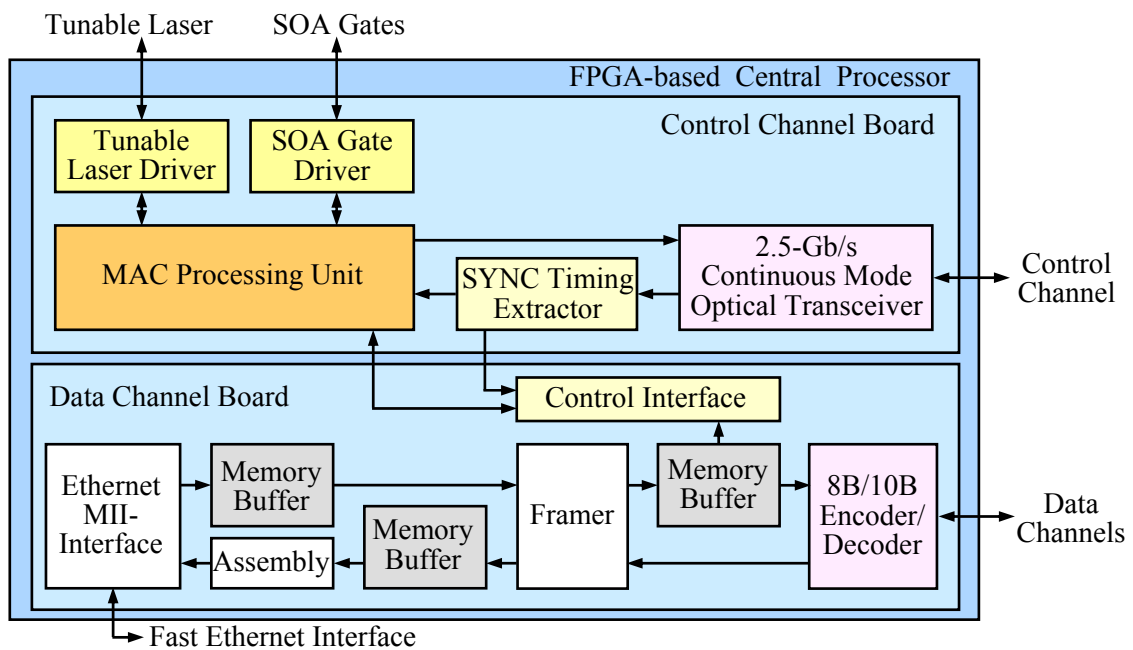


(a) Experimental node setup (S-node)



Legend: PBS: Polarization beam splitter OBPF: Optical bandpass filter

(b) Four-wave-mixing-based fast tunable filter/receiver



(c) FPGA-based central processor

Figure 13. Hardware implementation of the HOPSMAN testbed system.

Besides a fast tunable transmitter, as shown in Figure 13(a), a node (S-node) contains three major components: an FPGA-based central processor, a fast tunable filter/receiver, and an optical slot eraser. These components are described in detail in the following sections.

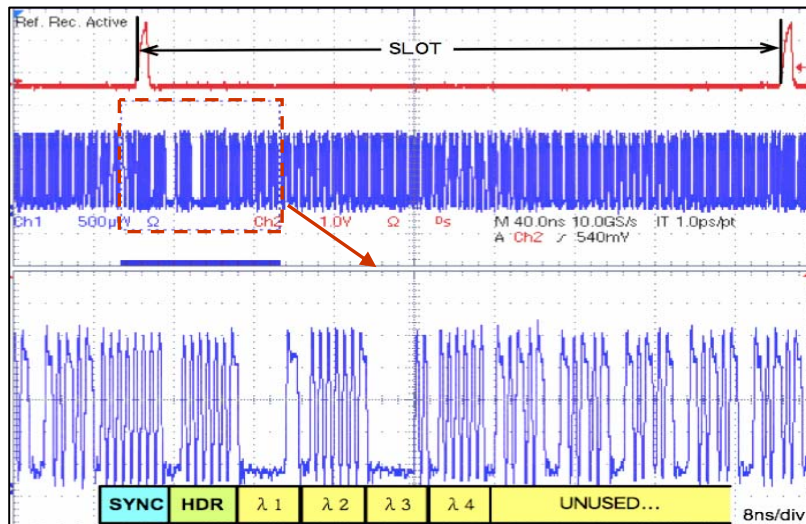
3.4.1 Channel Synchronization and Medium Access Control

The FPGA-based central processor consists of a control-channel board and a data-channel board, as shown in Figure 13(c). The processor is responsible for performing four major functions: channel synchronization, medium access control, optical device control, and data packet framing. Before describing these functions, we address a number of key design features for channel synchronization on HOPSMAN. For WDM slotted-ring networks, the timing synchronization between the data and control channels must be perfectly maintained at all times. In the HOPSMAN testbed, the channel-timing synchronization is ensured via two levels of alignment, which are coarse-grained and fine-grained synchronization, as well as guard-time-based dispersion compensation.

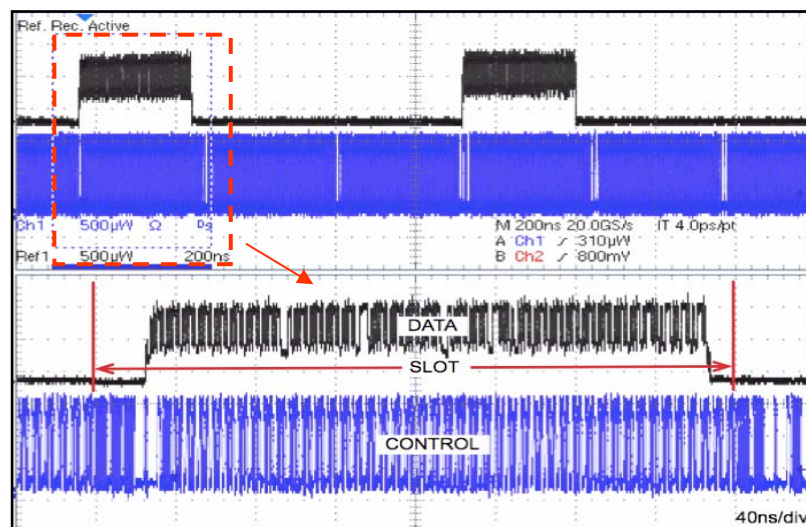
First-level coarse-grained synchronization is achieved by inserting a fixed short-fiber delay line (5 m in our case) in the optical data-channel path to accommodate the basic control computation latency. The second-level fine-grained synchronization is accomplished by matching a fixed-pattern preamble field (i.e., the SYNC field) at the beginning of each control slot, as shown in Figure 14(a). Moreover, as a result of the fiber's inherent chromatic dispersion, after long fiber transmissions the pre-aligned data channels undergo different propagation delays and are no longer synchronized with the control channel. For HOPSMAN's ring length of less than 50 km, simply adding a guard-time field at the beginning and/or end of each data slot can solve the problem. In the HOPSMAN testbed, the data can be correctly recovered

without any error with a guard time of 40 ns. HOPSMAN's data and control slots were found to be perfectly synchronized, as shown in Figure 14(b). Note that longer rings require an in-line dispersion compensation module to tolerate the propagation-delay difference.

The control-channel board contains a Xilinx VertexII FPGA chip and a 2.5-Gb/s continuous-mode optical transceiver. It is responsible for the first three functions (i.e.,



(a) Control channel slot



(b) Synchronized data and control slots

Figure 14. Synchronization of control and data channels.

synchronization, access and device control) of the central processor. Initially, the optical transceiver strips off the control slot from the ring. Each control slot (Figure 14(a)) contains one 16-bit SYNC field, one 16-bit header, and four 16-bit mini-slots, respectively, carrying the states of four data channels. The SYNC timing extractor (STE) mainly detects the SYNC field in the control slot. Upon having matched the SYNC field, the STE passes the precise timing trigger to the data-channel board via the control interface to bring the output data slot into full alignment with the control slot.

Followed by the STE, in accordance with the status of each data channel, the MAC processing unit (MPU) performs the MAC scheme, namely PQOC, which includes the five operations described next. Each data slot has four distinct states—BUSY, BUSY/READ (BREAD), IDLE, and IDLE/MRKD (IMRKD):

(1) To transmit a packet from the memory buffer into an IDLE slot on a wavelength, the MPU signals the tunable laser driver to perform the wavelength tuning, and updates the state from IDLE to BUSY in the corresponding mini-slot.

(2) To receive a packet from a wavelength, the MPU directs the same wavelength-tuning operation, but updates the slot state from BUSY to BREAD.

(3) To erase a BREAD slot, the MPU of an S-node informs the slot eraser module via the SOA gate driver in the control-channel board.

(4) As a result of having no packet in the memory buffer but with positive quota, the MPU yields an IDLE slot to downstream nodes (and earns a credit) by changing the state from IDLE to IMRKD.

(5) Thus, with a credit, the MPU transmits a packet from the memory buffer into an IMRKD slot by changing the state from IMRKD to BUSY. Finally, the updated control slot is sent back to the ring through the optical transceiver.

The data-channel board contains a Xilinx Spartn3A FPGA chip. It is responsible

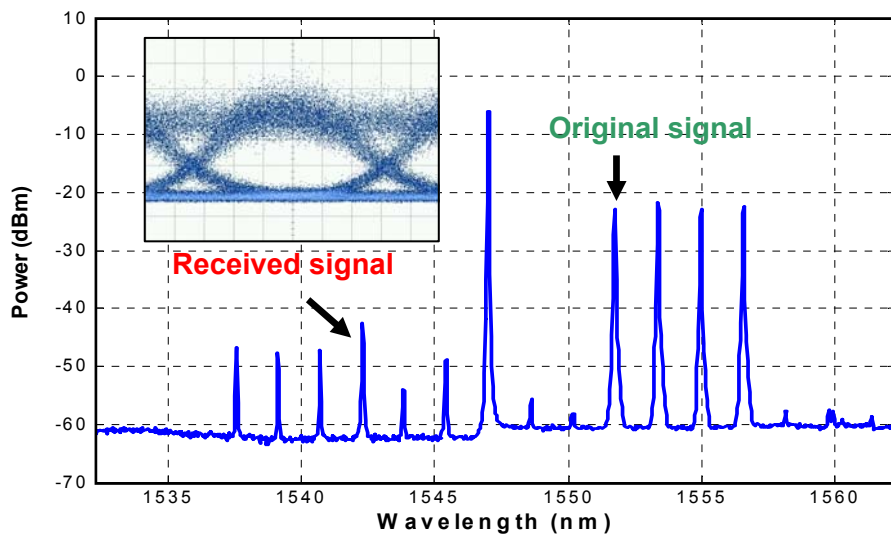
for the last function of the central controller, namely data-packet framing between the Fast Ethernet and the HOPSMAN ring. Note that the testbed can support any type of local area network and interface; we use Fast Ethernet only because of its wide availability. Specifically, for the outbound flow, the framer module first segments incoming Ethernet packets into smaller 350-bit-long HOPSMAN slots. Before being sent to the ring, data packets are encoded via the 8B/10B encoder, which enables reliable transmission and easier burst-mode reception. In the inbound flow, the framer performs the reverse function by assembling a number of data slots back to an original Ethernet frame.

3.4.2 Fast Tunable Filter/Receiver and Optical Slot Eraser

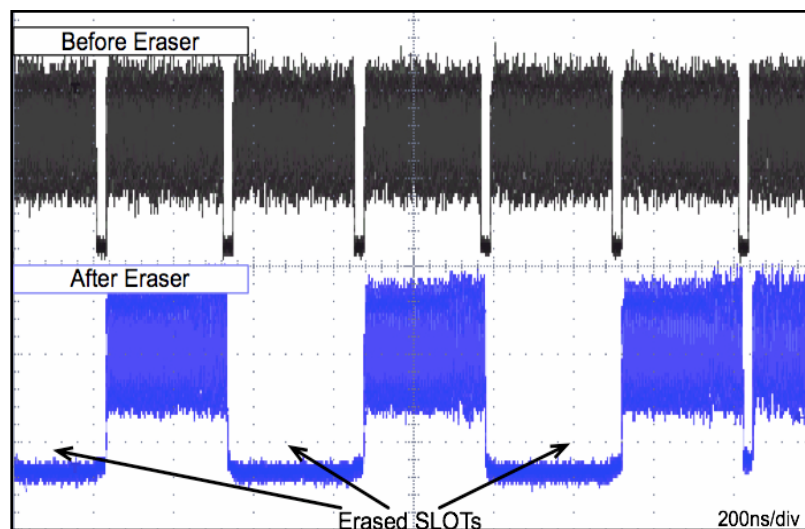
Tunable filters made from mechanically moving elements usually require millisecond tuning times, which is not feasible for optical packet-switching networks. New devices that achieve tuning times on the order of a microsecond have been proposed. Among them, the electro-optic tunable filter (EOTF) [71] can achieve sub-microsecond tuning speed, but requires a high tuning voltage. The acousto-optic tunable filter (AOTF) [72] reaches microsecond speed but only during the selection of channels. The fiber Fabry-Perot-based tunable filter [73] also efficiently provides a response time of up to few microseconds. In principle, the microsecond-level tuning time is still unacceptable for an OPS network that adopts a slot as small as 320ns, as HOPSMAN does.

In the HOPSMAN testbed system, we adopted a polarization-insensitive four-wave-mixing (FWM)-based optical tunable filter/receiver, as shown in Figure 13(b). Based on the FWM method, by using a sampled-grating distributed-Bragg-reflector (SGDBR) tunable pumping laser and an SOA, the wavelength of the tapped-off data signal can be converted to the target wavelength,

namely the wavelength of the fixed filter provided. The inherent polarization-tracking problem of this FWM-based system is solved using polarization diversity [49], as illustrated in Figure 13(b). The approach attains a conversion efficiency of 18 dB. Since the system tuning time depends on the tuning speed of the pumping laser, our FWM-based tunable filter/receiver achieves a tuning time of less than 25 ns. The



(a) Received signal by four-wave-mixing-based filter/receiver



(b) Waveforms before and after optical slot eraser

Figure 15. Experimental results with fast optical devices.

experimental result in Figure 15(a) displays the optical spectrum and the eye diagram of the received signal.

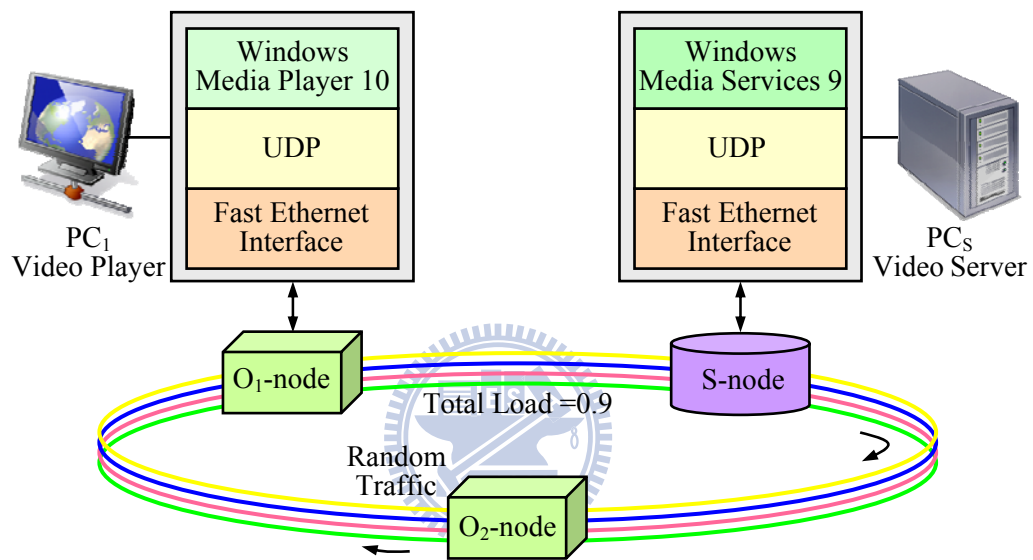
The optical slot eraser was built with a Mux/Demux pair and an array of SOA gates, which can be turned on/off in 5 ns and achieve an on/off extinction ratio greater than 30 dB. Figure 15(b) displays the two distinctive waveforms of a data channel before and after the erasing operation. The SOA gates also provide a 10-dB gain to cover the nodal loss contributed by the control-channel add/drop filter and Mux/Demux filters.

3.4.3 Demonstration of a Commercial Real-Time Application

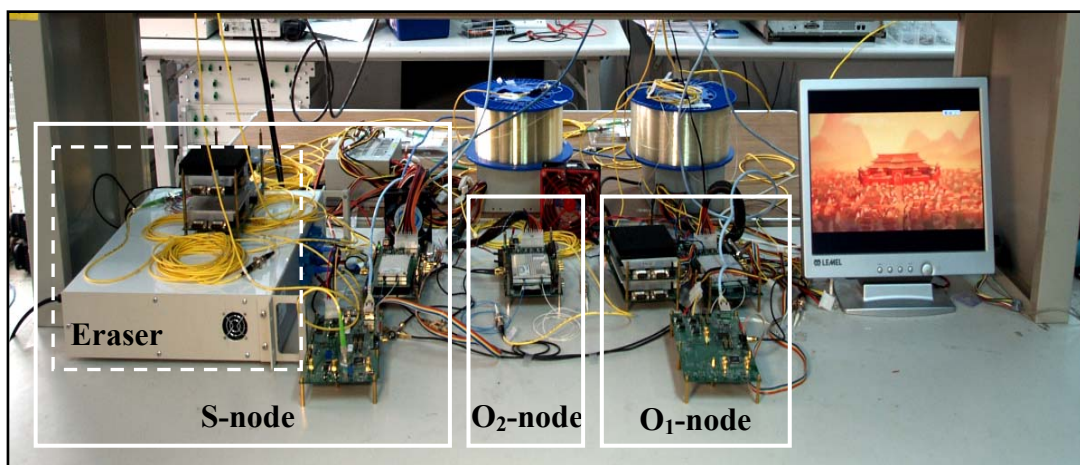
We conducted a feasibility test by running commercially available remote-media-player applications over a four-wavelength HOPSMAN testbed, as shown in Figure 16. There are three nodes in the testbed: S-node, O₁-node, and O₂-node. The S-node and O₁-node are connected to PC_S and PC₁, respectively, via a Fast Ethernet interface. At PC₁, a video-playback application, Windows Media Player 10, requests a thirty-minute-long 5.2-Mb/s MPEG-4-encoded video stream to be sent from PC_S, which runs a video-server application, Windows Media Services 9. The third node of the testbed, i.e., O₂-node, serves as a mass traffic generator, continuously sending dummy traffic to both O₁-node and S-node. The total amount of traffic to be generated is determined according to the following guidelines: the normalized per-wavelength load is set as high as 0.9, and the real-time video-stream traffic occupies only 0.25% of the total load ($0.9 \times 4 = 3.6$). In other words, the video-stream traffic is only provided with 0.25% of quota for the entire bandwidth.

Based on our simulation results, the maximum normalized throughput of the network with only a single server is 0.667. Accordingly, the maximum achievable throughput for the 8B/10B-encoded video-stream traffic is

$1.25\text{-Gb/s/wavelength} \times (8/10) \times 4 \times 0.9 \times 0.667 \times (0.25\%) = 6 \text{ Mb/s}$. As a result of poor bandwidth allocation, such a setting makes HOPSMAN a potential bottleneck for the video-stream traffic. With the PQOC scheme, the testbed has been shown to achieve delay- and jitter-free video playback at PC₁ in the O₁-node. With experiments of this sort, we have concluded that HOPSMAN is particularly advantageous for



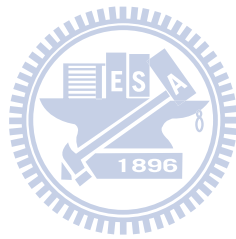
(a) HOPSMAN testbed: experimental setup



(b) A snapshot of the HOPSMAN testbed system

Figure 16. Feasibility test and demonstration of HOPSMAN testbed.

bandwidth-hungry and delay/jitter-sensitive applications. Medical imaging, on-line interactive gaming, distance learning, and remote terminal services, are among potential applications for HOPSMAN.



Chapter 4. MAC Scheme with QoS Assurance

In this section, we present a MAC scheme, called Probabilistic Quota plus Credit with QoS Assurance (PQOC/QA). PQOC/QA allows HOPSMAN to efficiently support various types of traffic, providing connection-oriented services for real-time traffic. The real-time traffic, which we will refer to interchangeably as high priority data in the remainder of the thesis, includes CBR (constant bit rate) and VBR (variable bit rate) traffic. Similarly, the data traffic, also called the low priority data, represents ABR (available bit rate) traffic.

4.1 Design Principles and the Detailed Algorithm

4.1.1 Slot-Basis Reservation

We first observe an access constraint imposed by having limited hardware resources in WDM network. Since each node has only one tunable receiver, receiver-contention occurs when there is more than one packet destined for the same receiver in a single slot time. Thus, two packets that are destined for the same node are prohibited to simultaneously occupy a single slot time via two different wavelengths. Likewise, because a node has only one tunable transmitter, any node is restricted to access at most one wavelength in a single slot time. Such a limitation is referred to as the vertical-access constraint. Because of this constraint, the transmission of real-time traffic is not guaranteed by adopting rate-basis reservation scheme as most MAC schemes in single-channel ring network. In the WDM ring network, the system may run into a situation where an excessive number of packets (including low and high priority data) are destined to the same destination node. When this situation is combined with the receiver-contention problem, the transmission of high priority data to some node may be hindered even though they are

already rate-reserved. To guarantee the real-time traffic transmission in WDM network, instead of rate-basis reservation, we propose slot-basis reservation to both resolve the receiver-contention problem and to satisfy QoS requirements.

PQOC/QA implements the slot-basis reservation simply through a marking mechanism at the control slot. We then illustrate the control slot format- each control slot contains a header (for synchronization purpose), as well as W mini-slots that carry the destination addresses, statuses and reservation fields of the corresponding W data slots (see Figure 17). There are four distinct states for each data slot- BUSY, BUSY/READ (BREAD), IDLE, and IDLE/MRKD (IMRKD). The statuses not only indicate whether the corresponding data slot is empty or busy, but they also function as notifications to the other nodes. For example, the BREAD status presents that the data has been received by the destination node, so the status also notifies the next S-node to erase the data slot of the corresponding wavelength and alter the status back to IDLE so that the slot can be reused. IMRKD, on the other hand, is a status that informs the downstream nodes of any bandwidth that still remains from any unused quota of upstream nodes. Furthermore, the data slot is reserved for high priority data

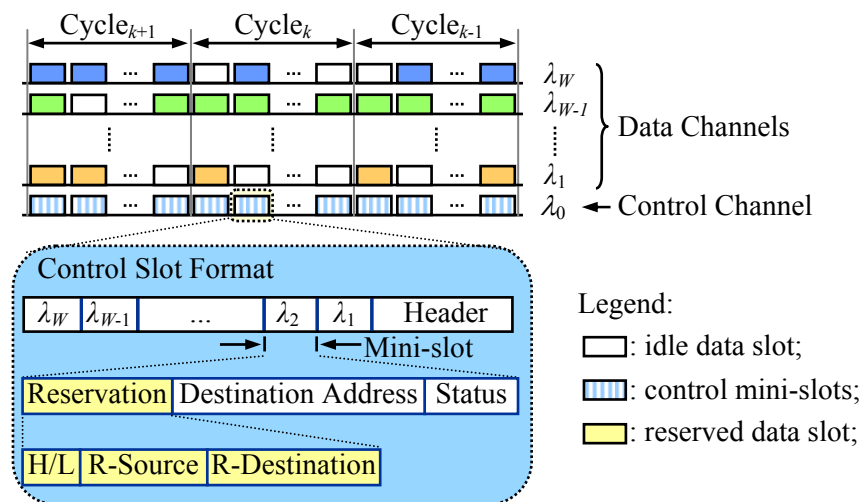


Figure 17. Cycle and slot and reservation structures.

by marking and also filling with the source-destination addresses of the connection at the reservation field of the corresponding mini-slot. The source-destination addresses of the connection helps prevent other nodes from sending data to the same destination at the same slot time, thereby solving the receiver-contention problem. Also, one should notice that since there is only one transmitter, the source node is forbidden to make more than one reservation in the same slot time. Clearly, this marking mechanism complies with the vertical-access constraint.

Specifically, PQOC/QA propose a connection-oriented scheme to support QoS, and the connection is established by marking on each cycle of the ring. Regardless of CBR or VBR connections, we mark the bandwidth based on the mean rate of each connection, which we will refer to as “mean rate reservation” afterwards. In other words, by slot-basis reservation, each connection is guaranteed to transmit its mean rate number of packets per each cycle, which is referred to as “mean rate transmission” in the following descriptions.

4.1.2 Call Admission Control

In order to satisfy QoS requirements, one of the key requirements for the call admission control (CAC) scheme is to ensure the accommodation of the fluctuated VBR traffic. We propose a straightforward, efficient, and distributed CAC scheme. Figure 18 shows the quota distribution in PQOC/QA. In essence, the CAC in a node locally admits a new real-time connection only if the total amount of the mean rates of the accepted CBR/VBR connections and the new connection is bounded under a predefined quota ratio, i.e. $r_H \cdot Q$, where $0 < r_H \leq 1.0$. The other proportion of quota, $(1 - r_H) \cdot Q$, is the minimum guaranteed number of quota to transmit the extra VBR traffic (the arrivals of this connection except the number of packets within its mean rate) and ABR traffic as well, while the extra VBR traffic has higher priority over the

ABR traffic. In other words, the scheme has a guaranteed proportion of the bandwidth left over for the bursty VBR traffic and ABR traffic. If the quota ratio is set reasonably, the probability that the extra VBR traffic fails to transfer due to expired quota is significantly small. Consequently, PQOC/QA and along with the simple CAC function, the system can achieve extremely low delay and jitter for VBR traffic of various burstiness. Moreover, as a connection is allowed to setup, the number of the quota which is equivalent to the mean rate of the connection is considered to be used. Then, the remaining quota, which is equal to Q minus the total amount of the mean rates of the accepted connections, is all left for transmitting extra VBR traffic and ABR traffic. Consequently, the CAC function is regarded as a simple and flexible scheme to facilitate bandwidth allocation. More importantly, it is also as a methodology to accommodate VBR traffic fluctuation.

We then discuss two important aspects regarding to the CAC scheme. First, we decide to reserve the bandwidth by mean rate of the connections rather than other schemes, such as effective bandwidth. Despite the simplicity for quota distribution, more important of all, mean rate reservation is more suitable than effective bandwidth reservation in our system. For example, if we mark the bandwidth based on the

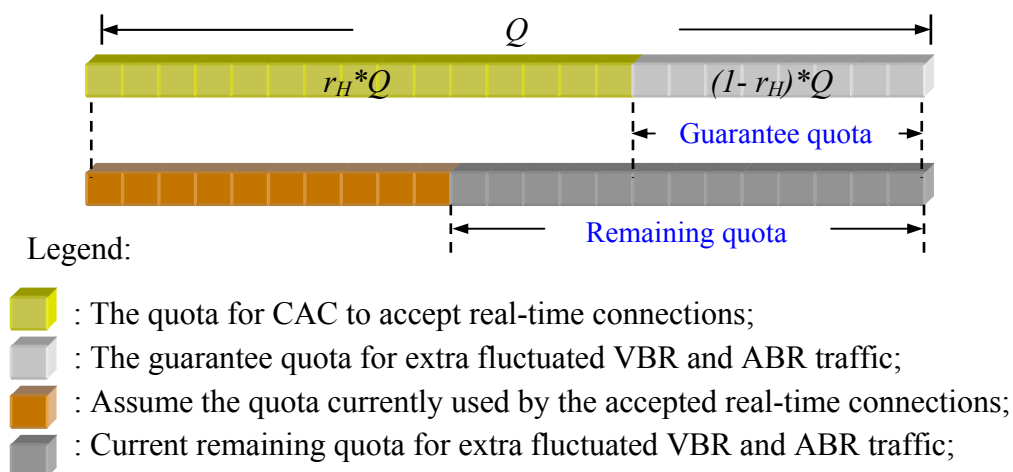


Figure 18. Quota distribution in PQOC/QA.

effective bandwidth (e.g. twice the mean rate), the overestimation may cause many bandwidths to be wasted and induce the system utilization degradation. Additionally, the maximum number of acceptable real-time connections is also more limited than mean rate reservation. Therefore, instead of estimating the real requirements of VBR traffic, we consider use the constant mean rate reservation methodology. Combining with the quota distribution, PQOC/QA not only assures QoS but also help remain the overall system utilization. Another important aspect is to determine the value of r_H . Notice that there is a system tradeoff involving setting the value of r_H . For example, the bigger the value of r_H , the bigger guaranteed load of real-time traffic. While the smaller the value of r_H , the better QoS guaranteed. Actually, while the network is with real-time traffic of high aggregate burstiness, a smaller value of r_H is set to assure the desired service requirements. However, once the statistical fluctuation of aggregated real-time traffic is of low burstiness, a bigger value of r_H can be chosen to achieve bigger load of real-time traffic while still assure the QoS requirements.

4.1.3 Call Marking Stage – Call Setup

In the call marking stage (at call setup), the node marks the required number of slots in each cycle of the ring. Once the markings are made successfully in all cycles on the ring, the connection is established and ready to transmit the real-time data. We consider two issues in the marking stage. First, the marked bandwidths have to satisfy the vertical-access constraint and promise to be available after the connection is setup ok. Second, the mean rate transmission must be guaranteed. In the first issue, to satisfy the vertical-access constraint, the node chooses those slot times which are without transmitter-contention (this node has not marked at those slot times) and without receiver-contention (other nodes have never marked to the same destination at those slot times). To assure the marked bandwidths to be available for transmission

after a single ring time, the node chooses IDLE, IMRKD and BREAD slots to mark because these slots are either empty, or will be erased by an S-node that it will encounter next. Therefore, the marked bandwidths are well prepared after a single ring time when the connections may be successfully established and ready for transmission.

To guarantee the mean rate transmission, we observe the data flow of the connections. The data flow varies with the relative locations of the source and destination nodes. To specify the locations of the nodes, we assume that there are S server-nodes (S-node 1 to S-node S) in the network dividing itself into S sections (sections 1 to S), with each section containing more than one node (including the server node of the section). Due to the geometry of a single ring, we observe that when downstream nodes send data to upstream nodes in the same section, the data will circulate around the ring for more than a single ring time. Therefore, the data still occupies the marked slot when it reaches the source node again, causing the node to fail to transmit another data at the following ring time. To guarantee the mean rate transmission, the node marks twice the mean rate of the connection in each cycle when downstream nodes send data to upstream nodes in the same section; while in the other cases, the node marks the number of mean rate of the connection in each cycle. In the marking stage, if the node fails to mark at any one of cycle, the node will attempt to mark in another 2 or 3 rounds of ring times. If these attempts still fail, the node is responsible to unmark those marked slots and block the connection.

In the following, we will illustrate the operation for establishing connections and also explain the data flow of the connections. First, the operation for establishing connections can be best explained via a simple example illustrated in Figure 19. Note that the quota in this example is not computed from the given network parameters, instead, to briefly describe the connection establishment, we arbitrarily choose a value

of quota, $Q = 4$. Accordingly, CAC accepts at most 2 ($=r_H \cdot Q$) quota for high priority data. Since the mean rates of the connections are of 1 slot per cycle, so each of them will use one quota. Therefore, connections c1, c2 are allowed to setup, while connection c3 waits in the queue (the quota for high priority data is expired, so c3 waits until c1 or c2 complete its transmission and releases the quota). Based on the relative locations of the source and destination nodes of the connections, c1 will mark 2 slots per cycle and c2 will mark 1 slot per cycle. To describe clearly, the cycles on the ring are indexed from Cycle₁ to Cycle₄. At slot time 4 (at Cycle₁), the connection c2 fails to make reservation due to the receiver-contention problem with other reservation. The reservations repeat until the end of one ring time (at Cycle₄), and the connection c1 is set up successfully and is ready for transmission in the following cycle. However, the connection c2 has been failed to mark at Cycle₁, so it continues to mark at the next cycle (Cycle₁), and also successfully establishes the connection. We further note in this example, if connection c2 failed to mark at other Cycle (e.g.

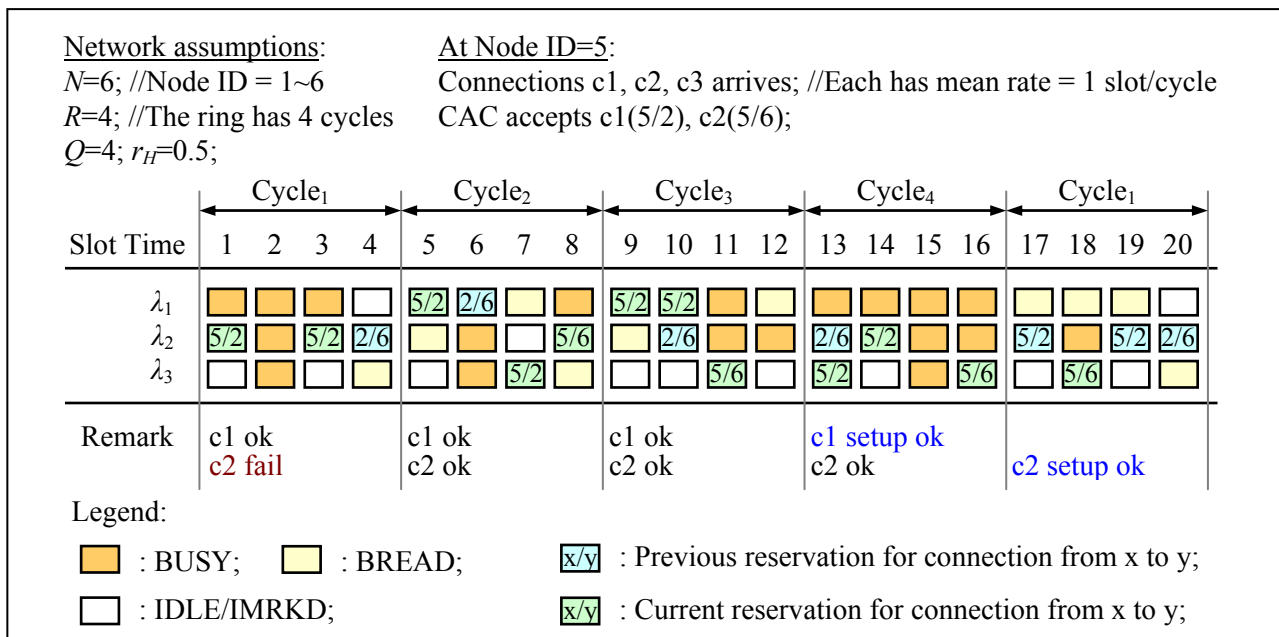
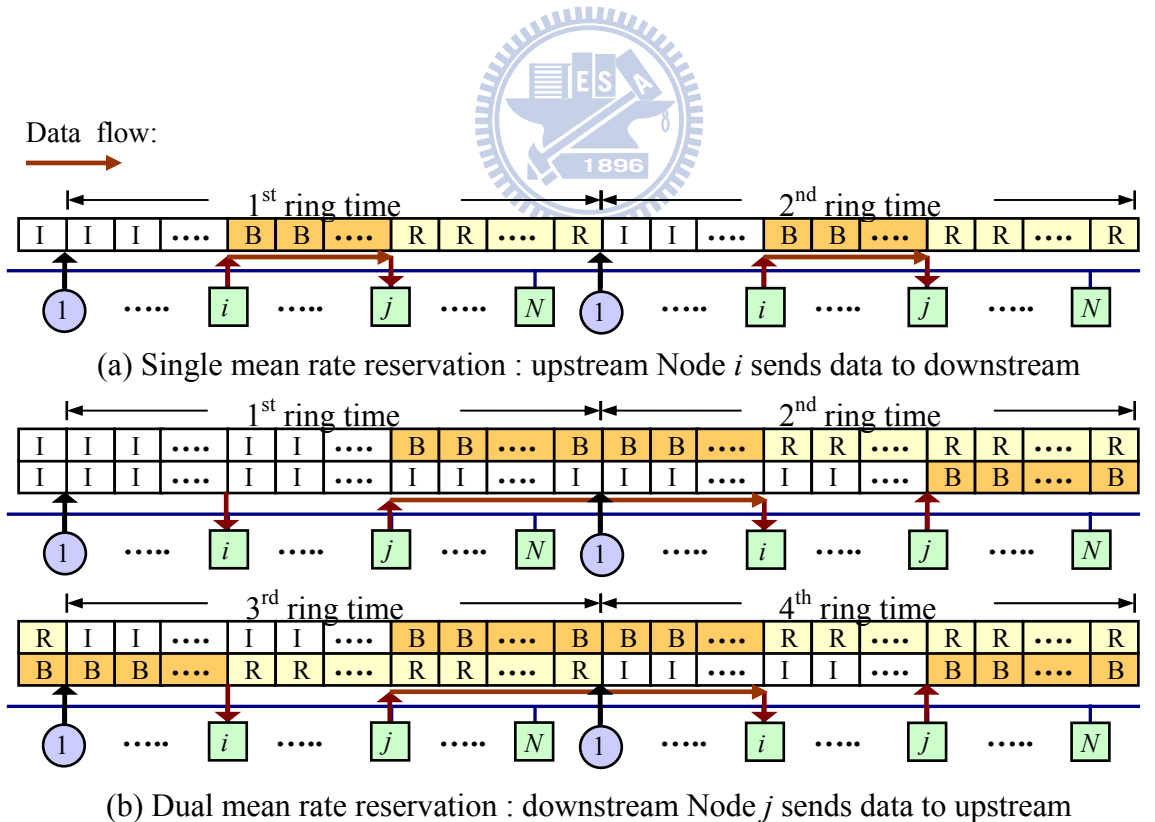


Figure 19. An example of connections set up.

Cycle₂), the connection c2 will wait until the failed cycle comes and endeavor to mark at that cycle again.

Further, the data flow of the connections is well illustrated in Figure 20. We simply assume that there is only 1 S-node ($S=1$), which is denoted as node 1. The data flow direction is from node 1 to node N , and repeats from node 1 to node N again because of the ring architecture. In Figure 20(a), upstream node i has established a connection to downstream node j by single mean rate reservation. In the 2nd ring time, we observe that node i successfully finds empty marked bandwidth to transmit data. In Figure 20(b), downstream node j has established a connection to upstream node i by dual mean rate reservations, which is presented as two marked bandwidth in each vertical line. Notice that in the 2nd ring time, node j will find a BREAD and an empty



Legend:

: Server node (node 1); : Reserved bandwidth (mean rate per cycle);
 : Ordinary node (node 2~N); : BUSY; : BREAD; : IDLE/IMRKD;

Figure 20. Data flow of the real-time connections.

marked bandwidth in each cycle. We observe that if we just make single mean rate reservation, node j will just find a BREAD marked bandwidth, thereby failing to send traffic for the whole ring time (This will happen at the 2nd, 4th ... ring times). Clearly, the description of the data flow verifies the correctness of the single and dual mean rate reservation.

4.1.4 Integrated Transmission Stage

In the transmission stage, we propose an integrated transmission policy for ABR, CBR, and VBR traffic. The scheme first transmits real-time traffic if the node finds its own marked slots. Otherwise, the node transmits extra VBR traffic and ABR traffic (which are termed as unreserved traffic) by the remaining quota. Certainly, the extra VBR traffic has higher priority over the ABR traffic. The implementation of the integrated transmission is fairly simple and described in the following paragraph.

At each slot time, each node first checks whether any mini-slot (in the corresponding control slot) contains its own reservation or not. If it finds its reservation, then the node will transmit a packet from the corresponding source-destination connection (CBR/VBR) and update the state from IDLE to BUSY. Otherwise, the node tries to transmit a packet from unreserved traffic. When the node wishes to transmit the unreserved traffic, it must gain access permission, which is determined by the probabilistic quota [8] and its success in finding an IDLE slot. If an IDLE slot is found, the node transmits the packet and updates the state from IDLE to BUSY. On the other hand, the node accumulates the number of transmission allowance to transmit the unreserved traffic in the next available slot (if without its own marked slot at that slot time). A destination node that has successfully dropped a packet modifies the slot state from BUSY to BREAD. This allows the next S-node to erase the data slot by changing the status from BREAD back to IDLE, thus enabling

slot reuse by downstream nodes. Also, if a node has no unreserved traffic to transmit but attains access permission based on the probabilistic quota, the node will earn the remaining number of slots as credits for future use. Once this happens, the node will update the statuses of the same number of data slots from IDLE to IMRKD. Finally, a node uses its credits to transmit more unreserved traffic beyond the probabilistic quota on any IMRKD data slots within the window, and subsequently updates the state to BUSY. This integrated transmission policy seamlessly combines the connectionless and connection-oriented traffic.

Furthermore, to maximize the aggregate system throughput, PQOC/QA also uses bandwidth flexibly. For example, if a node finds a VBR connection that has fewer arrivals than its mean rate in a cycle, the unused marked slots will be used by ABR traffic instead. On the other hand, if the node does not have any proper ABR traffic to transmit at that slot time, the credit accumulates accordingly and the bandwidth can be used by downstream nodes. In essence, the system may have some marked but unused bandwidths during call setup, or because of the burstiness arrival of VBR traffic. Either the node itself, or downstream nodes can take advantages of these marked bandwidths. The nodes are allowed to choose the unreserved traffic, including ABR or extra VBR traffic. However, the packets should be carefully chosen to prohibit subsequent transmissions of the original connections from being obstructed; that is, the packets must be erased by an S-node before the source nodes of the original connections can use them. Although quite many traffic can not be transmitted by this bandwidth, in our scheme, this kind of unused bandwidth is very limited and is often possible to be temporarily used. Recall the scheme that marks bandwidth by effective bandwidth has bigger marked bandwidth than our scheme, so the probability that no proper traffic can temporarily use the marked bandwidth is much bigger, thereby resulting in system utilization degradation. Consequently, with this flexible use of

bandwidth and the mean rate reservation scheme, the aggregate system throughput can still be optimized when the network has QoS traffic.

4.1.5 The Detailed Algorithm

The PQOC/QA algorithm for real-time traffic is given in Figure 21. The algorithm is executed on a per-slot basis. If the slot is specified as the beginning of a cycle, the algorithm executes a simple CAC function to determine whether the node accepts new connection(s) or not. The CAC function is implemented by comparing the sum of the mean rate of a new connection ($c.r$) and the total amount of the reserved high priority data (nh) with the quota for high priority data ($Q_H = r_H \cdot Q$). If the summation is lower than the quota bound (Q_H), the new connection is allowed to enter into the marking stage (call setup), and nh is concurrently added with the value of the mean rate of the new connection. The node sets up a connection by marking required bandwidth (single/dual mean rate: $c.r/2 * c.r$) in each cycle of the ring. The slots are chosen to be marked wherever the status is not BUSY at those slot time without transmitter- or receiver- contention. Then, the transmitting stage is taken place after the connection establishment, and the connection will finally be torn down after the transmission completion (the node is responsible to unmark those marked slots).

4.1.6 Efficiency of Mean Rate Reservation

In essence, the PQOC/QA adopts a simple and straight methodology to support QoS. In the following, we will prove that our methodology is the best solution in the network system. In our scheme, the CAC accepts new connections if the following equation stands,

$$\sum_i m_i \leq r_H \cdot Q, \quad (9)$$

where m_i is the mean rate of each accepted connection. Let us consider another scenario: we reserve the bandwidth by effective bandwidth. For example, the effective bandwidth is twice the mean rate [45]. (Some researches have derived a bigger number of effective bandwidth.) Therefore, in this scenario (called EB-scenario afterwards), the CAC accepts new connections if the following equation stands,

$$\sum_i 2m_i \leq Q, \quad (10)$$

which can be re-written as-

$$\sum_i m_i \leq 0.5Q. \quad (11)$$

Equation (9) and equation (11) are very similar, except that r_H is replaced by 0.5. As will be described in the simulation results, our scheme can achieve great performance even under r_H is set as high as 0.8 or 0.9. In other words, our scheme is possible to accept a larger number of real-time connections than the EB-scenario (if r_H is set above 0.5). More importantly, the EB-scenario reserves a bigger number of bandwidths, resulting in having a bigger possibility to waste bandwidth. As described, the number of ABR traffic which can flexibly take advantages of those reserved but unused bandwidth is limited. Since our scheme adopts mean rate reservation, the number of reserved but unused bandwidth is also quite limited. However, due to over-reservations, the EB-scenario causes system utilization degradation. Besides, our proposed scheme also provides a simple bandwidth allocation. As a result, although with simple design, it is the best and most suitable methodology to apply in our system.

Variables

MYID: the node address;

Q_H : quota for high priority data; ($= \lfloor r_H \cdot Q \rfloor$)

c : a connection;

nh : the sum of currently accepted high priority data;

Slot type

Header : {CYCLE_BEGIN, NORMAL};

Status : {BUSY, BREAD, IDLE, IMRKD};

Priority: {L, H (R_{src} , R_{dest})};

Rsrc: source address of a reserved connection;

Rdest: destination address of a reserved connection;

Connection data type

$c.dest$: connection destination;

$c.r$: transmit rate (slots / cycle);

$c.phase$: {WAIT, SETUP, TRMIT, RELEASE};

Initialization(c)

$c.phase =$ WAIT;

if (MYID **and** $c.dest$ are in the same
network section **and** MYID > $c.dest$)
plan to reserve $2 \cdot c.r$ slots per cycle;

else plan to reserve $c.r$ slots per cycle;

endif

Main() /*execute at each slot time*/

1. Add new connections arrivals c to a queue;

Initialization(c);

2. Read the control slot;

3. **if** (slot's *Header* is CYCLE_BEGIN)

for (each c **and** $c.phase =$ WAIT)

if ($nh + c.r \leq Q_H$) //permit to setup

Update WAIT to SETUP;

$nh += c.r$;

endif endfor endif

4. **for** (each c)

switch $c.phase$:

case SETUP: Setup(c); **break**;

case TRMIT: TX_Hdata(c); **break**;

case RELEASE: Release(c); **break**;

endfor

Figure 21. Detailed PQOC/QA algorithm.

Setup(*c*)

```
if (find a mini-slot is H with  $R_{src} = MYID$ )  
    No reserve; endif; //transmit contention  
else if (find a mini-slot is H with  $R_{dest} = c.dest$ )  
    No reserve; endif; //receive contention  
else if (not reserve completely in this cycle and find an L mini-slot is not BUSY)  
    Update L to H ( $MYID, c.dest$ ); //reservation  
endif  
if (reserve ok on all the cycles of the ring)  
    Update SETUP to TRMIT;  
else if ( time out )  
    Update SETUP to RELEASE; endif
```

TX_Hdata(*c*)

```
if (not transmit completely in this cycle and find an empty mini-slot is H ( $MYID, c.dest$ ))  
    Transmit data of c; endif  
if (c transmits all)  
    Update TRMIT to RELEASE; endif
```

Release(*c*)

```
if (not release completely in this cycle and find a mini-slot is H ( $MYID, c.dest$ ))  
    Update H to L; endif  
if (release all the reservations on all the cycles of the ring)  
    Remove c from queue;  $nh -= c.r$ ; endif
```

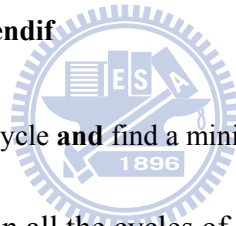


Figure 21. Detailed PQOC/QA algorithm (continue).

4.2 Expected Queueing Time Analysis

In our analysis of the expected setup queueing time, the system is modeled as an M/G/m queueing system. The M/G/m queueing system contains m servers with Poisson arrival and a general service time. More specifically, the new CBR/VBR connections arrival process is assumed to be a Poisson process with parameter λ . Also, the number of servers is equal to the maximum admissible quota of real-time traffic. Our goal is to accurately determine the expected queueing time for an M/G/m queueing system with a particular general service distribution (to be described later) in our system. In this analysis, we will calculate the occupancy distribution under the ordinary service discipline, the first-come-first-served (FCFS) discipline. Using the result of the occupancy distribution, we will then obtain the expected number of customers in the queue. Finally, by applying Little's formula, we acquire the expected queueing time for the M/G/m system.

Before we begin our analysis, we must first define the service distribution to be used in our M/G/m queue. Consider the case that each connection is with mean rate of 1-slot per cycle, that is, each connection use one quota (considered as a server) when it is allowed to be set up. Contrastingly, when a connection is torn down, the used quota is released. Therefore, the service time is measured from the beginning of the connection setup until the time when the transmission is complete. In other words, the service time includes the time spent in call marking stage (referred to as setup marking time) and the transmission time of the connection. As will be explained in Section 4.3, the setup marking time is almost equal to one ring time. Based on this fact, we can assume that the setup marking time is a constant value. We can also assume that the transmission time of a connection is simply exponentially distributed. Consequently, the service time is an exponential duration added over a constant value,

and follows the general distribution $f_{\tilde{x}}(x)$ defined below-

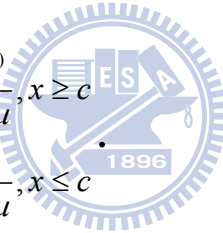
$$f_{\tilde{x}}(x) = \mu e^{-\mu(x-c)}, x \geq c . \quad (12)$$

Here, \tilde{x} is the service time random variable, while μ is the mean rate of the exponential distribution, and c is a constant value. Note that a Metropolitan Area Network (MAN) ring has a typical length between 50 and 200 km. If we consider a signal propagation speed of 2×10^8 m/s, one ring time is approximately 250 μ s to 1 ms. It is straightforward to demonstrate that the duration of a single connection is more than just a few times longer than a single ring time. In other words, the mean length of the exponential distribution is at least a number of times greater than the constant value. We term this sort of service distribution as similar to an exponential distribution. The derivation strategy begins with the determination of the remaining service times found by arrivals and departures, which are primarily resolved by the following two facts and one observation.

The first fact states that the remaining times for the customers in service found by the Poisson arrivals or left by Poisson departures are mutually independent and identically distributed in the same way as the distribution of the residual service-life variables [74]. And, the second fact states that the stationary departure process is a Poisson process if and only if the service distribution is exponentially distributed [75]. The second argument presents an obstacle in deriving the remaining work at departure instants for other general service distributions. Fortunately, we observe that if the general service distribution is similar to an exponential distribution then the departure process will be similar to a Poisson process. Since the particular service distribution in our system is similar to an exponential distribution, we make an assumption in our approximation that the remaining service times left behind by a departure approximately follow the distribution of the residual service-life. With these facts and

observation in mind, and using the requirements of residual service-life, we then subsequently calculate the occupancy distribution as in the following paragraphs.

We define \tilde{x}_i^r , ($1 \leq i \leq m$) as the forward recurrence time or residual life for the service time, \tilde{x} . Actually, $\tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{m-1}^r, \tilde{x}_m^r$ are independent random variables, each with the residual life distribution $f_{\tilde{x}_i^r}(x) = (1 - F_{\tilde{x}}(x)) / E[x]$. Here, index i indicates the number of the residual service-life variables and should not be mistaken for the index of any particular server. Because these random variables are all mutually independent and identically distributed, \tilde{x}_i^r does not have to belong to any specific server. Further, using the given service distribution as in Equation (12), the remaining service distribution \tilde{x}_i^r can be derived to be-

$$f_{\tilde{x}_i^r}(x) = \frac{1 - F_{\tilde{x}}(x)}{E[x]} = \begin{cases} \frac{e^{-\mu(x-c)}}{c + 1/\mu}, & x \geq c \\ \frac{1}{c + 1/\mu}, & x \leq c \end{cases} \quad (13)$$


Then, we examine the equilibrium probability in the system. Let P_j^a be the equilibrium probability that an arriving customer finds j customers in the system; and let P_j^d be the equilibrium probability that a departing customer leaves j customers in the system. In the M/G/m queueing system, the stationary departure and arrival distributions are equal for systems where arrivals and departures occur one by one [76]. Clearly, $P_j^a = P_j^d$. We also cite a well-known result in queueing theory, Poisson arrivals see time average (PASTA), which states that the stationary arrival distribution is equal to the stochastic equilibrium occupancy probabilities ($P_j^a = P_j$). From the reasons provided above, it is not difficult to see that the arrival, departure, and

stochastic equilibrium occupancy distributions are all identical in an M/G/m queueing system. This explains why we denote these probabilities in the same way in the following derivations; i.e., $P_j = P\{j \text{ customers in the system at an arbitrary point in time, or at an arrival or departure instant}\}$.

Figure 22 is a visual aid that we will continue referring to in the following derivation. First, we assume that a virtual arrival (who arrives in the system at time t_v but does not require any service) finds j ($j > 0$) customers in the system. Prior to this virtual arrival, there are two possible previous events: the previous event may be an arrival that sees $j-1$ customers in the system (see Figure 22, Event A), or a departure that leaves j customers behind (see Figure 22, Event D). We denote the time elapsed between the arrival (Event A) or departure (Event D) instant and the beginning of the next service completion epoch as \tilde{z}_A and \tilde{z}_D , respectively. Let \tilde{t} denote the interarrival time, an exponential random variable with parameter λ . Note that the next service completion times \tilde{z}_A, \tilde{z}_D are definitely greater than \tilde{t} . Otherwise, another event (a departure) would happen before t_v , which contradicts the assumptions made earlier. Since we have argued that the arrival, departure, and stochastic equilibrium occupancy distributions are identical, so it yields

$$P_j = P_{j-1} \cdot P(\tilde{t} < \tilde{z}_A) + P_j \cdot P(\tilde{t} < \tilde{z}_D), \quad (14)$$

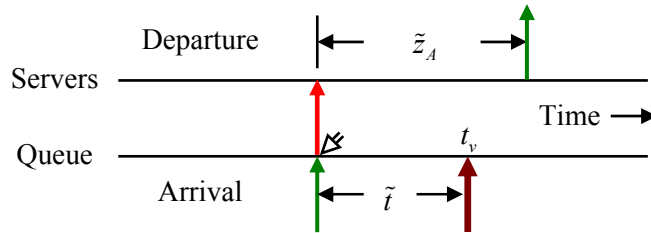
where

$$P(\tilde{t} \leq \tilde{z}) = \int_0^{\infty} (1 - e^{-\lambda z}) dF_{\tilde{z}}(z) = 1 - F_{\tilde{z}}^*(\lambda), \quad (15)$$

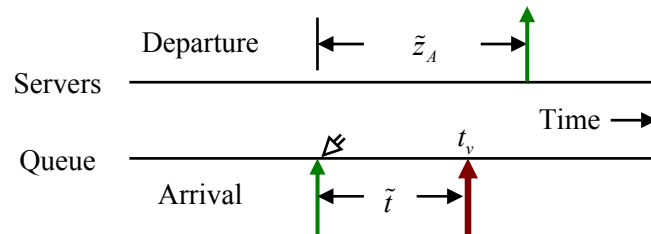
and $F_{\tilde{z}}^*(\lambda)$ is the Laplace Transform of \tilde{z} with parameter λ , where \tilde{z} can be replaced by \tilde{z}_A or \tilde{z}_D . With just the fact and approximation assumption that the arrivals and departures find the customers in service with the remaining service times distributed as \tilde{x}_i^r , we can compute the occupancy distribution.

Event A: An arrival sees $j-1$ customers in the system

Case A1: An arrival enters a server ($j \leq m$)

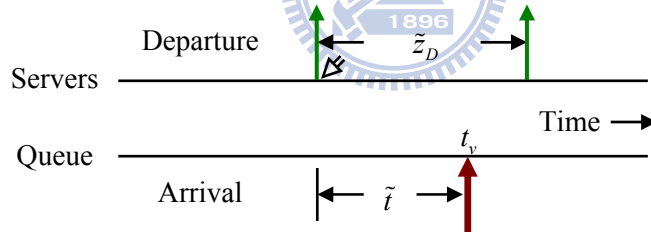


Case A2: An arrival waits in the queue ($j > m$)

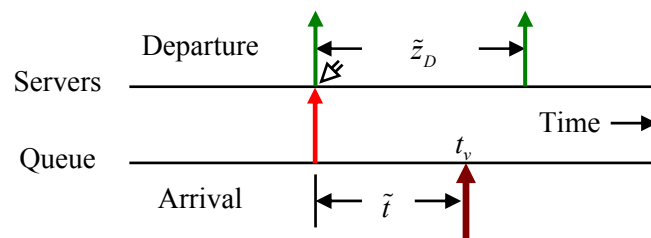


Event D: A departure leaves j customers in the system

Case D1: No customer in the queue ($j < m$)



Case D2: A queueing customer enters a server ($j \geq m$)



Legend:

\nless : The event epoch; \tilde{t} : Inter-arrival time;

t_v : The arrival time of the virtual customer;

\tilde{z}_A : The time between the instant of event A and the next service completion epoch;

\tilde{z}_D : The time between the instant of event D and the next service completion epoch;

Figure 22. Occupancy distribution analysis for M/G/m under FCFS.

The derivation is presented in the following pages. When an arrival comes seeing $j-1$ customers in the system, it either directly enters an idle server or waits in the queue (because all servers are occupied). This causes the service time \tilde{z}_A to be distributed as $\min\{\tilde{x}, \tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{j-1}^r\}$ and $\min\{\tilde{x}_1^r, \tilde{x}_2^r, \tilde{x}_3^r, \dots, \tilde{x}_m^r\}$ (see Figures 22, Case A1 and A2), respectively. When a departure occurs, it either leaves j customers who are already in service or releases a server for the first customer in the queue. Thus, the service time \tilde{z}_D is distributed as $\min\{\tilde{x}_1^r, \tilde{x}_2^r, \tilde{x}_3^r, \dots, \tilde{x}_j^r\}$ and $\min\{\tilde{x}, \tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{m-1}^r\}$, (see Figures 22, Case D1 and D2), respectively. By taking this, and Equations (14) and (15) into account, we can obtain the occupancy distribution P_j as shown in Equations (16):

$$P_j = \begin{cases} \frac{1 - F_{\tilde{x}}^*(\lambda)}{F_{\tilde{x}_1^r}^*(\lambda)} P_0, & \text{if } j = 1 \\ \frac{1 - F_{\min(\tilde{x}, \tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{j-1}^r)}^*(\lambda)}{F_{\min(\tilde{x}_1^r, \tilde{x}_2^r, \tilde{x}_3^r, \dots, \tilde{x}_j^r)}^*(\lambda)} P_{j-1}, & \text{if } 1 < j < m \\ \frac{1 - F_{\min(\tilde{x}, \tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{m-1}^r)}^*(\lambda)}{F_{\min(\tilde{x}, \tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{m-1}^r)}^*(\lambda)} P_{j-1}, & \text{if } j = m \\ \frac{1 - F_{\min(\tilde{x}_1^r, \tilde{x}_2^r, \tilde{x}_3^r, \dots, \tilde{x}_m^r)}^*(\lambda)}{F_{\min(\tilde{x}, \tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{m-1}^r)}^*(\lambda)} P_{j-1}, & \text{if } j > m \end{cases} \quad (16)$$

To proceed, we derive the necessary results:

$$F_{\tilde{x}}^*(\lambda) = \frac{\mu}{\lambda + \mu} e^{-\lambda c}, \quad (17)$$

$$F_{\tilde{x}_i^r}^*(\lambda) = \frac{1}{(\lambda + \mu)(c + 1/\mu)} e^{-\lambda c} + \frac{1}{\lambda(c + 1/\mu)} (1 - e^{-\lambda c}), \quad (18)$$

We must now assume that $\tilde{b}_1 = \min(\tilde{x}, \tilde{x}_1^r, \tilde{x}_2^r, \dots, \tilde{x}_{k-1}^r), k = 2, 3, \dots, m$. The probability density function (PDF) of \tilde{b}_1 is

$$F_{\tilde{b}_1}(b) = P(\tilde{b}_1 \leq b) = 1 - P(\tilde{b}_1 > b) = 1 - P(\tilde{x} > b, \tilde{x}_1^r > b, \tilde{x}_2^r > b, \dots, \tilde{x}_{k-1}^r > b)$$

$$= 1 - P(\tilde{x}_i > b) \cdot P(\tilde{x}_i^r > b)^{k-1} = \begin{cases} 1 - \left[\frac{e^{-\mu(b-c)}}{\mu c + 1} \right]^{m-1} \cdot e^{-\mu(b-c)}, & b \geq c \\ 1 - \left[\frac{\mu c + 1 - \mu b}{\mu c + 1} \right]^{m-1}, & b < c \end{cases}, \quad (19)$$

yielding

$$F_{\tilde{b}_1}^*(\lambda) = \int_0^\infty e^{-\lambda b} dF_{\tilde{b}_1}(b) \\ = \frac{ke^{-\lambda c}}{\mu^{k-2}(c+1/\mu)^{k-1}(\lambda+k\mu)} - (k-1)e^{-\lambda(c+1/\mu)} \int_1^{c\mu+1} w^{k-2} e^{\lambda w(c+1/\mu)} dw, \quad (20)$$

We must also assume that $\tilde{b}_2 = \min(\tilde{x}_1^r, \tilde{x}_2^r, \tilde{x}_3^r, \dots, \tilde{x}_k^r)$, $k = 2, 3, \dots, m$. The PDF of \tilde{b}_2 is

$$F_{\tilde{b}_2}(z) = P(\tilde{b}_2 \leq b) = 1 - P(\tilde{b}_2 > b) = 1 - P(\tilde{x}_1^r > b, \tilde{x}_2^r > b, \tilde{x}_3^r > b, \dots, \tilde{x}_k^r > b) \\ = 1 - p(\tilde{x}_i^r > b)^k = \begin{cases} 1 - \left[\frac{e^{-\mu(b-c)}}{\mu c + 1} \right]^m, & b \geq c \\ 1 - \left[\frac{\mu c + 1 - \mu b}{\mu c + 1} \right]^m, & b < c \end{cases}, \quad (21)$$

yielding

$$F_{\tilde{b}_2}^*(\lambda) = \int_0^\infty e^{-\lambda b} dF_{\tilde{b}_2}(b) \\ = \frac{ke^{-\lambda c}}{\mu^{k-1}(c+1/\mu)^k(\lambda+k\mu)} - ke^{-\lambda(c+1/\mu)} \int_1^{c\mu+1} w^{k-1} e^{\lambda w(c+1/\mu)} dw. \quad (22)$$

We can now substitute the numerical results of Equations (17), (18), (20), (22) into Equation (16). From the requirement that the probabilities sum to unity ($\sum_j P_j = 1$), we can derive the occupancy distribution (P_j). A final step involves letting \tilde{n}_q be the expected number of customers in the queue. By applying Little's formula, we find that $E[\tilde{w}] = E[\tilde{n}_q]/\lambda$, where $E[\tilde{n}_q] = \sum_{j=m}^\infty (j-m) \cdot P_j$. We have now obtained the expected queueing time of an M/G/m queue under an FCFS scheme.

4.3 Simulation Results

In this section, we will evaluate how PQOC/QA performs in two separate areas: 1) performance for connections setup; 2) performance for real-time and data traffic. The values used for the parameters in the simulation are listed below- the network has a total of 20 nodes ($N=20$), where node 1 is designated as an S-node. There are 21 cycles on the ring ($R=21$). Each cycle consists of 100 slots per wavelength ($C=100$). The optical fiber accommodates 9 wavelengths ($W=9$). Traffic destinations are assumed to be uniformly distributed among all nodes. In order to focus on the performance of real-time traffic, the number of server nodes is set to be 1 ($S=1$). The quota for each node is then calculated to be $Q = (2S/(S+2)) \cdot C \cdot W / N = 30$ [8]. The system load (L) represents the normalized load per slot per wavelength, including both a load of high priority data (L_H), and a load of low priority data (L_L). Clearly, $L = L_H + L_L$. Moreover, ABR traffic is directly generated to follow a Poisson distribution.

The real-time traffic is generated by the following parameters: the mean rate of a connection is 1 slot per cycle, the mean length of a connection is $\bar{l} + 2R$ slots ($\bar{l} = 2000$), the number of new connections arrivals has Poisson distribution with parameter $\lambda = Q \cdot L_H / (\bar{l} + 2R)$, and the aggregate burstiness of VBR connections is denoted as B . The parameters are taken according to the following principles. First, all the mean rates of the connections are simply assumed to remain at 1 slot per cycle, since a connection with a mean rate of x slots per cycle can be equally regarded as x number of connections with a mean rate of 1 slot per cycle. Second, we observe that if the length of a connection is shorter than a single ring time, we need extra management to tear down connections. Considering the high burstiness of VBR connections and the fact that the length of a connection is longer than a few number of

ring times, the length of a connection is set to be at least 2 ring times ($=2R$). Finally, the bursty arrival of VBR traffic is generated with a two-state (H and L) Markov Modulated Poisson Process (MMPP) [70]. More specifically, the MMPP is characterized by four parameters (α , β , λ_H , and λ_L), where α (β) is the probability of changing from state H (L) to L (H) in a slot, and λ_H (λ_L) represents the probability of arrivals at state H (L). Accordingly, if $\lambda_L = 0$, the mean arrival rate can be expressed as $\beta \times \lambda_H / (\alpha + \beta)$, while the aggregate burstiness (B) of VBR connections can be expressed as $B = (\alpha + \beta) / \beta$.

To simplify the simulations, each run of simulation contains only one type of high priority data, such as CBR traffic, or VBR traffic with an aggregate burstiness. Also, as described in the PQOC/QA algorithm, a maximum allocation ratio r_H ($0 < r_H \leq 1.0$), is proposed to constrain the maximum ratio bound of accepted high priority data. Therefore, we will only examine the non-overloaded simulation results, where $L_H \leq r_H$. The simulations are terminated after they reach a 95% confidence interval. We will elaborate on the significance of these results in the two following subsections.

4.3.1 Performance for Connections Setup

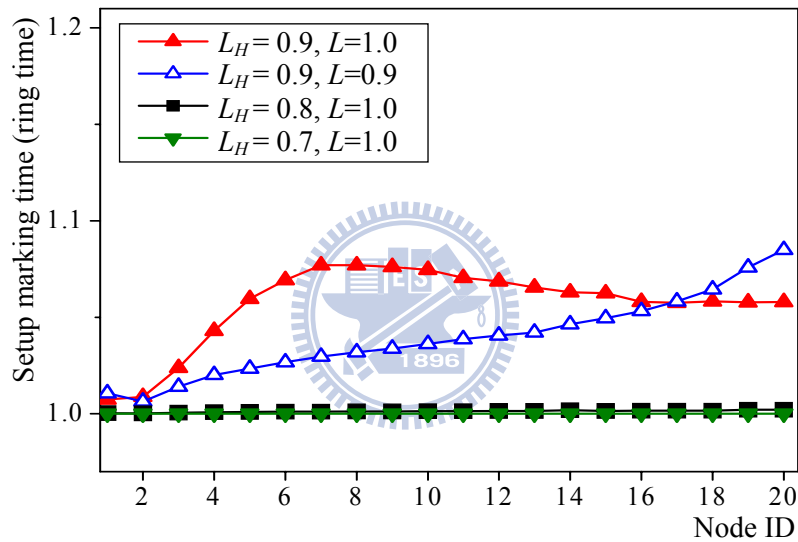
To evaluate the performance for connection setup, we first define the duration of a connection. The duration is the amount of time that has elapsed since the instant a connection is successfully setup and begins to transmit data, until all the data of the connection has been completely transmitted. Hence, the length of the duration is dependent primarily on the transmission rate and the length of a connection. Notice that because the arrival of a VBR connection is bursty, it is difficult for us to specify the exact duration. On the other hand, it is simple to specify the exact duration for a CBR connection, which is equal to the length of a connection when the mean rate is

1-slot per cycle. Therefore, the duration of a CBR connection will last an average number of $\bar{l} + 2R$ cycles in our simulation. Consequently, we will only consider CBR traffic when evaluating the connection setup queueing time. Also, the probability density function of the duration of a CBR connection is as in Equation (12), where \tilde{x} is the duration of a CBR connection, $\mu = 1/\bar{l}$, and $c=2R$.

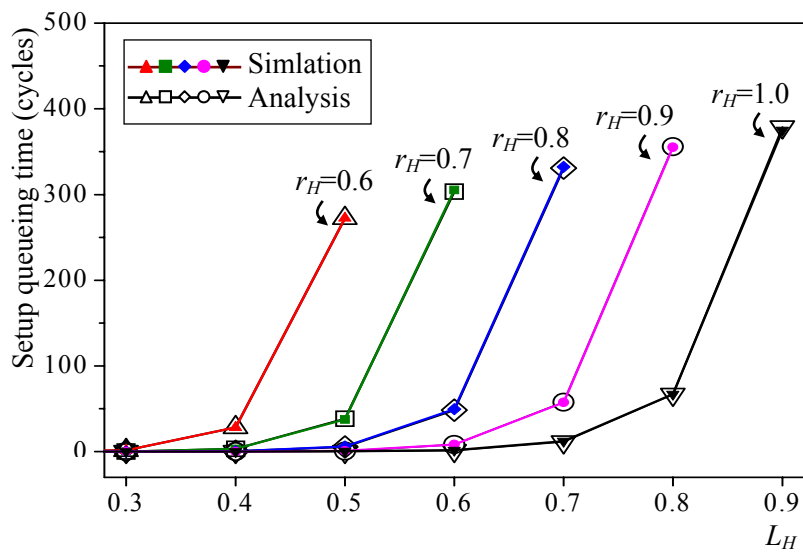
We will now discuss the time spent in call marking stage (referred to as setup marking time). To examine whether the connections can be successfully marked in the first ring time, we must compare the number of slots needed to establish new connections with the number of mark-able slots in a cycle. First, we compute the mean number of slots needed to establish new connections in a node. The first step is to make an approximation by considering the worst case scenario, where all new arrivals are allowed to set up their connections and to mark double spaces. Further assume the simple case that the setup marking time is exactly one ring time, thus we find that $2\lambda R$ slots would be needed. Next, we discuss what the mean number of mark-able slots in each cycle is. Recall that the mark-able slots include empty slots and BREAD slots. To simplify this discussion, we will not consider BREAD slots. Thus, each node sees at least $L_L \cdot Q$ empty bandwidths, which is a sufficient number for the call marking stage when $L_H \leq 0.9$ (i.e. $2\lambda R < L_L \cdot Q$). Actually, we have more than enough mark-able slots to use, especially when the lengths of the connections are prolonged for more than thousands number of cycles (which reduce λ). The results are verified in the simulation results as shown in Figure 23(a): the setup marking time is only 1 ring time under $L_H < 0.9$. While the setup marking time is a little higher than 1 ring time when $L_H = 0.9$, primarily due to the vertical-access constraint for such high loads. This causes the setup marking time to be almost equal to one ring time when $L_H \leq 0.9$. It is important to note that the network bounds the value of L_H by simply

setting the value of r_H . Further, we will analysis the waiting time before CAC accepts the call (setup queueing time). To make the analysis more straightforward, we will assume that the setup marking time is a constant value R .

We then validate the analytic results of our proposed expected queueing time of an M/G/m queue with the expected setup queueing time in our simulation. Consider a multiple-server queueing system, M/G/m, where the arrival process for new connections follow a Poisson process with parameter $\lambda = Q \cdot L_H / (\bar{l} + 2R)$, and the



(a) Setup marking time under various loads of high priority data



(b) Analytic and simulation results on setup queueing time

Figure 23. Connections setup performance.

number of servers (m) is set as quota for real-time traffic, i.e. $m = \lfloor r_H \cdot Q \rfloor$. We know from previous sections that the service distribution includes the constant setup marking time and the duration time of a CBR connection. The service distribution is as in Equation (12), where \tilde{x} represents the service time, $\mu = 1/\bar{l}$, and $c=3R$. By varying r_H and L_H , we obtain the expected setup queueing time from the simulation and analytical results. Figure 23(b) shows that all analytical results are in good agreement with the simulation results.

Finally, we would like to examine the mean setup queueing time. A quick look at Figure 23(b) shows that when r_H remains constant, the mean queueing time rises as L_H increases. On the other hand, if L_H remains the same, the queueing time decreases when r_H increases. This occurs because additional real-time traffic arrivals cause more connections to be queued, thereby enlarging the queueing time. Contrastingly, a greater r_H means that more real-time connections are accepted, thus reducing the queueing time. Evidently, the mean setup queueing time depends only on r_H and L_H .

4.3.2 Performance for Real-Time and Data Traffic

Here, we will first present the throughput, delay and delay bound performance of real-time traffic, and subsequently inspect the effect real-time traffic has on ABR delay. Figure 24 shows the simulation results of the throughput performance of high priority data. As depicted in Figure 24, the values of the maximum throughput of high priority data almost reach the values of r_H . As we know that r_H is the maximum ratio bound to accept the real-time traffic, so the maximum throughput can be approximated by a truncated Poisson distribution with parameter r_H , which is the reason why the maximum throughput is a bit lower than the value of r_H . Therefore, r_H is considered as a great and flexible mechanism for bandwidth allocation.

Furthermore, PQOC/QA flexibly transmits ABR traffic by using the remaining quota regardless of the value of r_H , and takes advantages of the reserved but unused bandwidth from upstream nodes. Though not depicted in the figure, PQOC/QA achieves exceptional aggregate throughput performance which is the same as previously proposed results when the network is only with ABR traffic (PQOC achieves 100% throughput under load is 0.9 or below [8]). Particularly, we inspect in the figure, when $r_H=1.0$, the maximum throughput is a bit better than PQOC under the condition when the network is highly saturated or overloaded. As previously discussed, when the network is highly saturated, PQOC has inevitable throughput deterioration for downstream nodes because of the vertical-access constraint. However, benefit from the slot-basis reservation, the constraint is somewhat lifted in PQOC/QA. As a result, PQOC/QA facilitates QoS differentiation but without causing the system throughput to degrade.

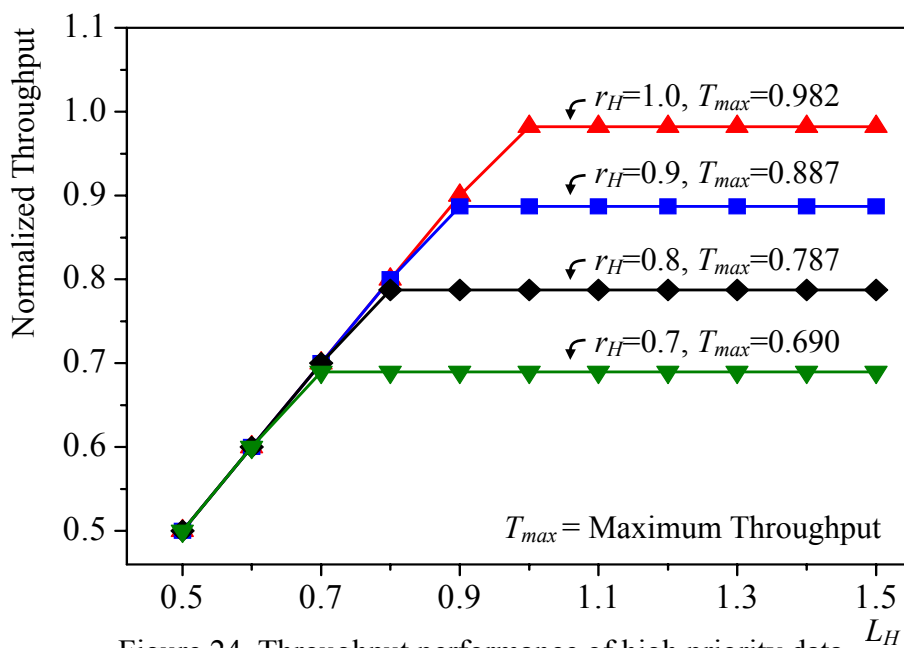
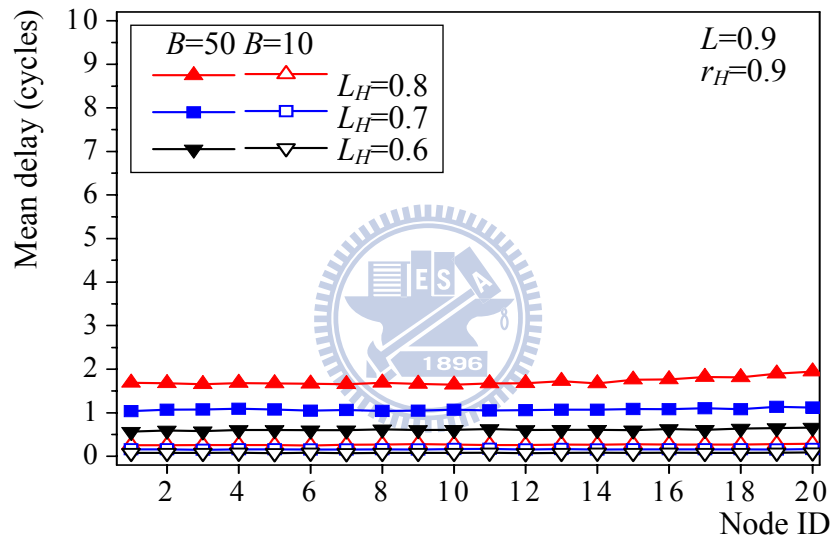
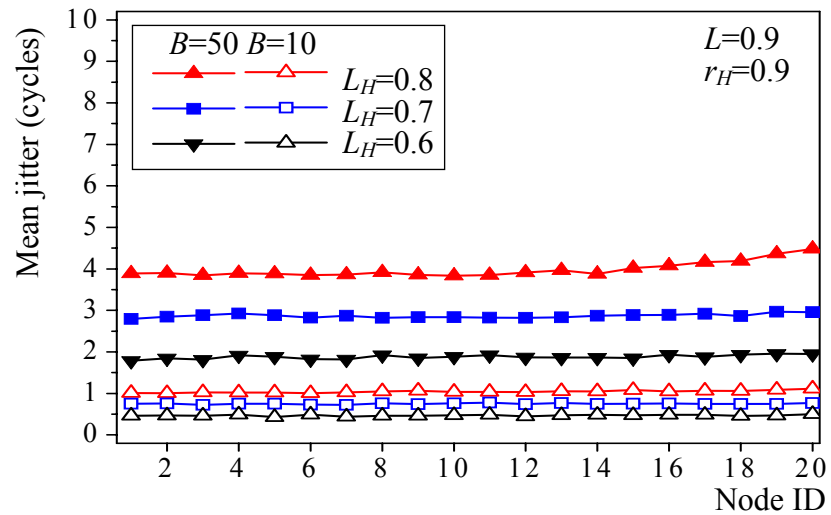


Figure 24. Throughput performance of high priority data. L_H

We now discuss in greater detail on the mean delay and jitter performance of real-time traffic in our system. The access delay of CBR traffic is zero because CBR traffic is sequentially transmitted to the marked slots. Therefore, we will only be considering VBR traffic here. The simulation results for VBR mean delay and jitter performance are depicted in Figures 25 and 26. As expected, the delay increases with the offered load and the burstiness of VBR traffic. As shown in Figures 25(a), (b), PQOC/QA guarantees delay/jitter fairness even when the value of r_H is set as high as



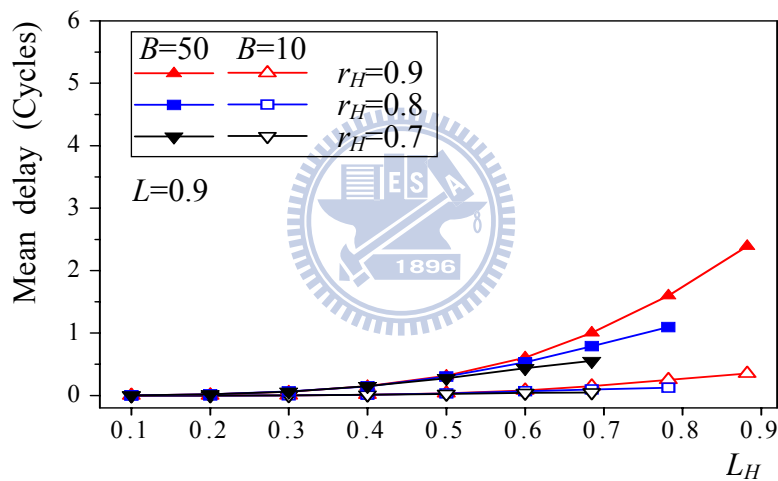
(a) VBR delay under various loads and burstiness



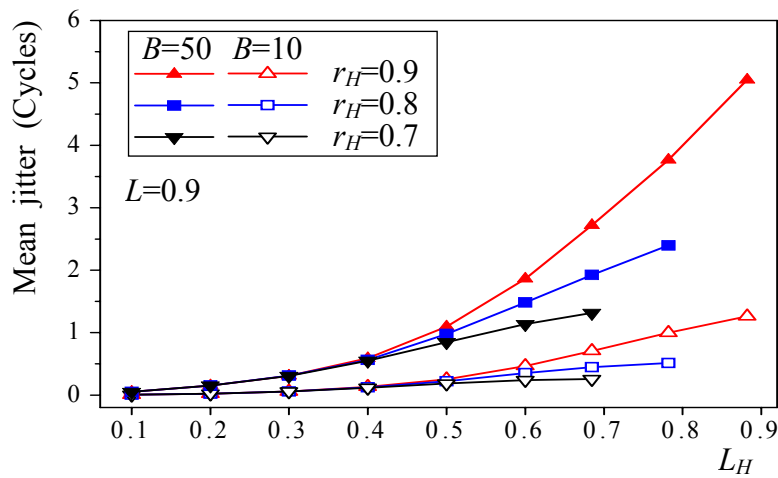
(b) VBR jitter under various loads and burstiness

Figure 25. Mean delay and jitter performance of VBR traffic.

0.9 and at high loads. The results also show that the scheme has considerably low VBR delay and jitter. For example, the results demonstrate almost negligible delay and very low jitter under a low burstiness ($B=10$) condition. Even in situations with high burstiness and heavy loads of high priority data, PQOC/QA still has very low delay and jitter. Furthermore, Figure 26(a), (b) show the mean VBR delay/jitter under various values of r_H . Observe that each curve runs the simulation results under all admissible values of L_H , where the maximum highest value of L_H is almost equivalent to r_H as indicated in Figure 24. As expected, delay/jitter increase with the value of r_H , since setting a smaller value of r_H will increase the minimum guaranteed proportion



(a) VBR delay under various r_H and burstiness



(b) VBR jitter under various r_H and burstiness

Figure 26. Mean delay and jitter performance of VBR traffic.

of bandwidth for extra VBR traffic. The results can be as an indication for the determination of setting the value of r_H .

Another important measurement is the delay bound of VBR traffic, as shown in Figure 27. Under a saturated system load ($L=0.99$), we draw 99.99%, 99.9%, 99% and 90% delay bounds. The bounds not only increase with the offered load and the burstiness of VBR traffic, but they evidently also increase with more stringent delay bounds. These results demonstrate that PQOC/QA assures low bounds under various loads and burstiness of VBR traffic. For example, even under the most stringent delay

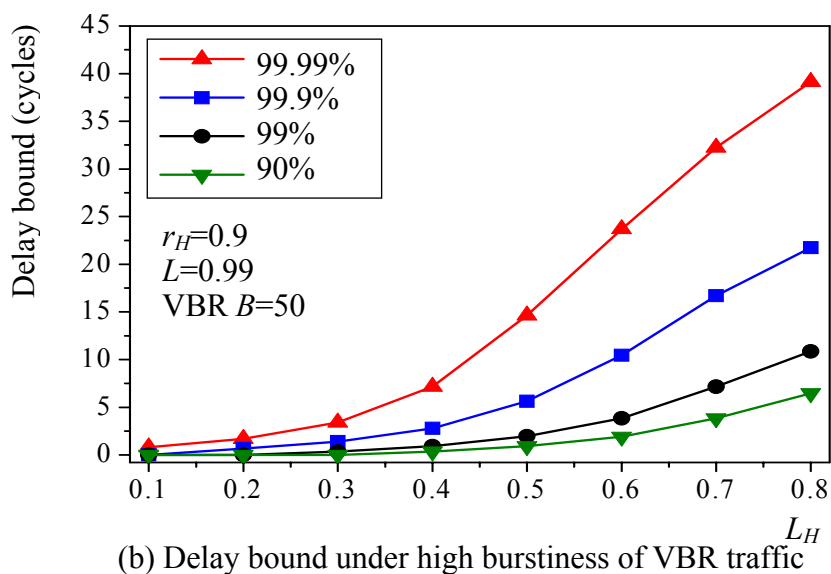
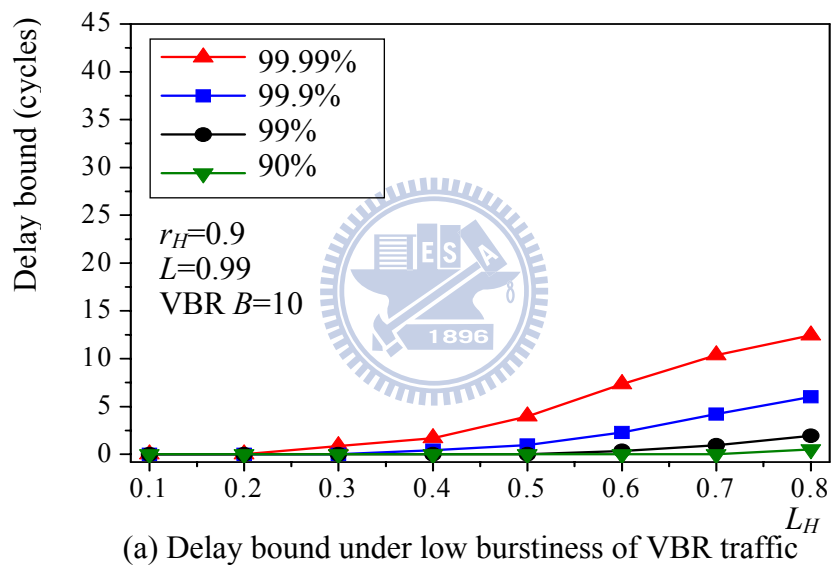
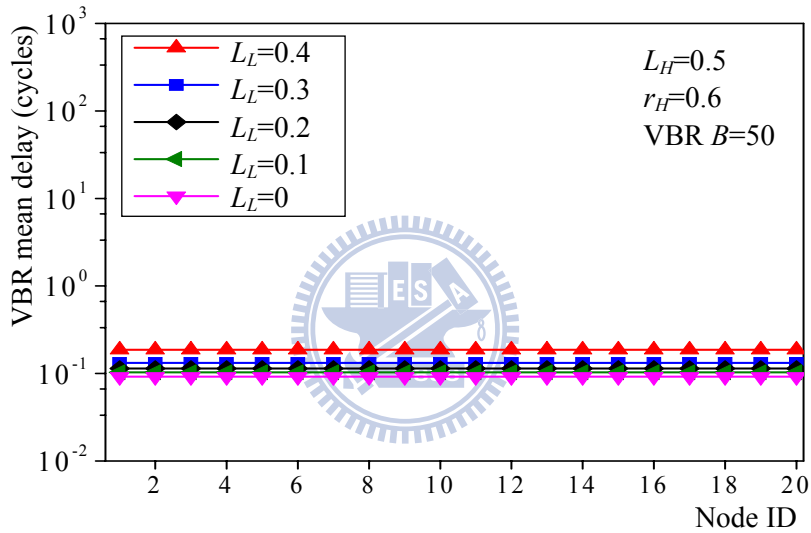


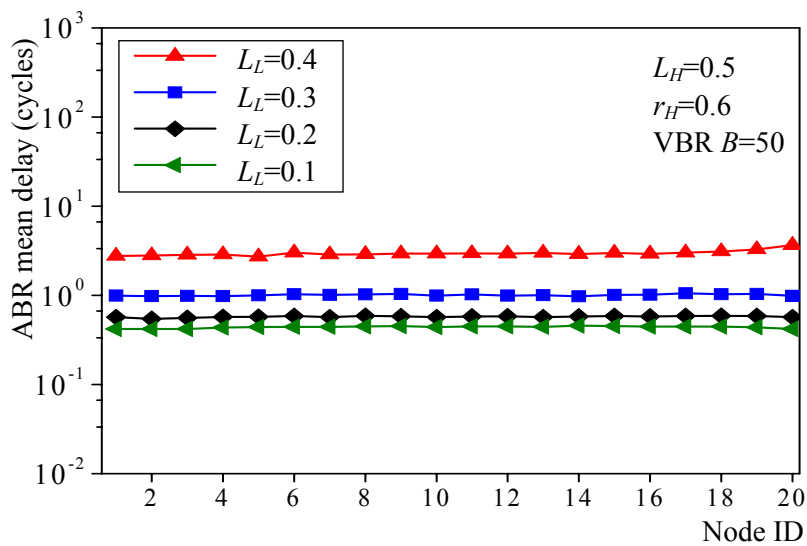
Figure 27. Delay bound of VBR traffic.

bound, 99.99%, with high burstiness traffic ($B=50$) at heavy load ($L_H=0.8$), the delay bound is only 40 plus cycles (which is equivalent to only 2 ring times). These results indicate that by strategically setting the value of r_H to constrain the maximum value of L_H , this system is capable of meeting various delay bound requirements of the network.

We can also observe the QoS differentiation of PQOC/QA by drawing a comparison between VBR and ABR delay. We first study the impact of ABR loads on mean VBR delay. As shown in Figure 28(a), the mean access delay of VBR traffic



(a) VBR mean delay under various ABR loads



(b) ABR mean delay under various ABR loads

Figure 28. The impact on VBR mean delay under various ABR loads and the ABR delay comparison.

increases only slightly with increasing ABR loads. In other words, the influence ABR loads have on VBR delay is negligible. Compared with the ABR delay in Figure 28(b), VBR traffic invariably assures low delay at the expense of slightly increased ABR delay. Another observation we can make from Figure 29 is that under equal loads ($L_H = L_L = 0.45$), ABR traffic suffers an obviously higher delay than VBR traffic. This delay is particularly pronounced when VBR traffic has higher burstiness ($B=50$). Overall, these results illustrate how PQOC/QA achieves low delay for VBR traffic and facilitates traffic differentiation.

We are now at a point where we can demonstrate how ABR delay is impacted by various real-time traffic loads. As shown in Figure 30, the mean access delay of ABR traffic first rises with increasing VBR traffic loads. This result should be expected since heavier real-time traffic will hinder the transmission of ABR traffic, thus increasing the delay. Rather surprisingly, however, ABR delay actually drops at the highest value of L_H for each curve under VBR traffic with $B=10$, as plotted in Figure 30(a). This occurs because the guaranteed quota, $(1-r_H) \cdot Q$, just happens to be enough for ABR traffic when $L_L \leq 1.0 - r_H$. On the other hand, when the system is placed under high burstiness of VBR traffic ($B=50$) (see Figure 30(b)), the impact on ABR delay increases and the drops for each curve disappear. Putting these special cases aside, the ABR delay is still mostly affected by loads imposed by real-time traffic. We observe that PQOC/QA maintains reasonable delay performance for ABR traffic if we choose a low to medium value of r_H .

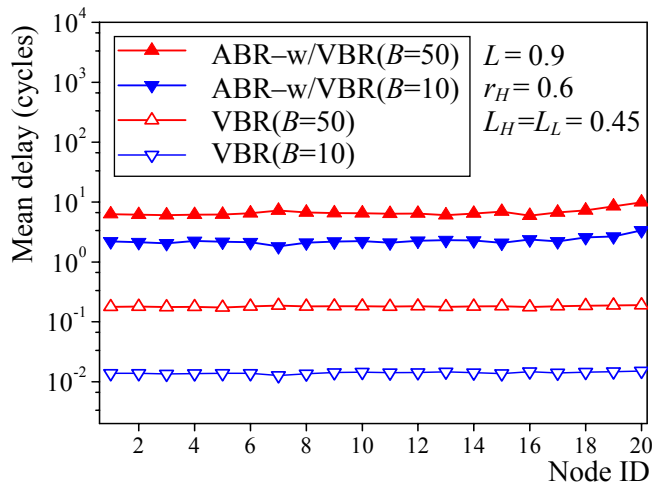
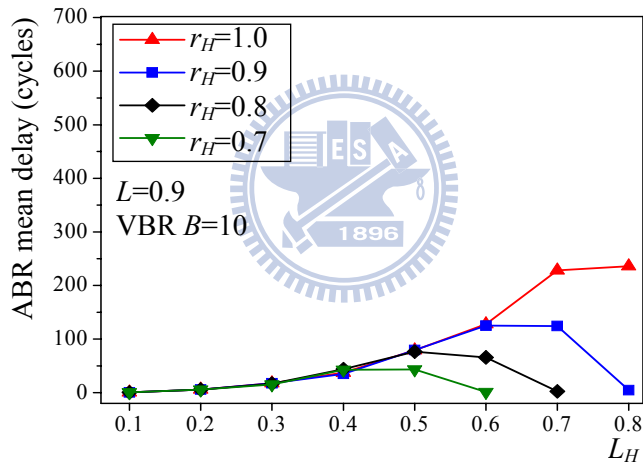
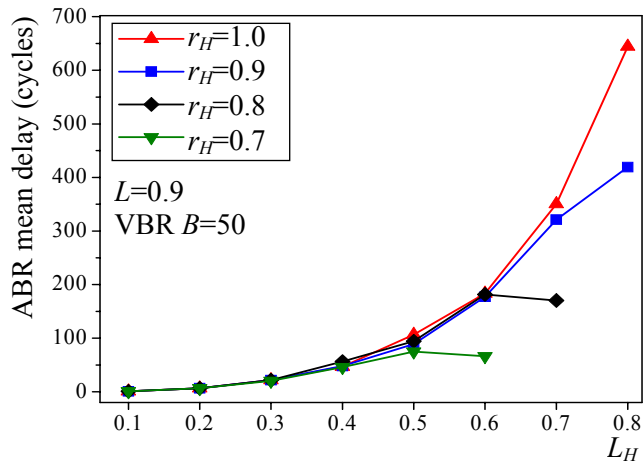


Figure 29. Mean delay comparison between ABR and VBR traffic under equivalent loads.



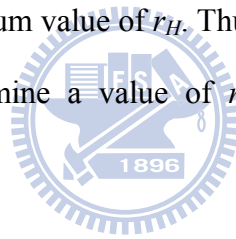
(a) ABR delay under low burstiness of VBR traffic



(b) ABR delay under high burstiness of VBR traffic

Figure 30. The impact on ABR delay under various burstiness and loads of VBR traffic.

Furthermore, we discuss how the value of r_H is determined. As described earlier, r_H operates as a bandwidth allocation scheme and as a mechanism to accommodate the VBR traffic fluctuation. Clearly, there appears to be a tradeoff that occurs when we set r_H to a larger or smaller value. A large value of r_H leads to a high bandwidth allocation for real-time traffic (a bigger guaranteed load of real-time traffic), thus contributing to a lower setup queueing time. For highly-bursty VBR traffic adaptation, a smaller value of r_H is set to assure the desired service requirements. However, once the statistical fluctuation of aggregated real-time traffic is of low burstiness, a bigger value of r_H can be chosen to achieve bigger load of real-time traffic while still assure the QoS requirements. Actually, an adequate value of r_H should be carefully chosen to satisfy diverse delay and delay bound guarantees. We also observe that the ABR delay also benefits from having a medium value of r_H . Thus, the simulation results can serve as guidelines to help us determine a value of r_H that achieves the best system performance.



Chapter 5. Conclusions

In this thesis, we have presented the architectural design, access control and hardware implementation of our experimental optical packet-switched metro WDM slotted-ring network, HOPSMAN. We proposed a novel medium access control which is called PQOC, and the MAC is then further enhanced with QoS assurance, called PQOC/QA. In addition to ordinary nodes (O-nodes), HOPSMAN encompasses a few server nodes (S-nodes) that are equipped with optical slot erasers, resulting in a significant increase in system throughput. Essentially, the MAC scheme employs a novel probabilistic-quota-based method to achieve fair and efficient bandwidth allocation. Given the number of S-nodes and destination-traffic distribution, we derived a closed-form formula for the determination of the probabilistic quota. The MAC scheme also uses a window-constrained credit-based approach to facilitate versatile allocation of the remaining bandwidth under highly-bursty and fluctuating traffic environments. Simulation results delineated that HOPSMAN achieves 100% throughput when there are only two S-nodes in the network. Furthermore, HOPSMAN with the MAC scheme was shown to achieve highly efficient and fair bandwidth allocation under various traffic loads and burstiness. HOPSMAN was justified robust and fair when under attack by malevolent nodes. Finally, the HOPSMAN testbed system uses FWM-based fast tunable filters/receivers and optical slot erasers that enable nanosecond-order optical packet switching operations. With flexible optical devices and an efficient MAC scheme (PQOC), HOPSMAN was shown, by means of a feasibility test, capable of achieving guaranteed delay-throughput performance particularly for bandwidth-hungry and delay/jitter-sensitive applications.

Furthermore, we propose the MAC scheme, PQOC/QA, which not only inherits

the original basic design of PQOC, but also integrate with QoS support on HOPSMAN. To support QoS and to resolve the intrinsic access problem in WDM network, PQOC/QA adopts slot-basis reservation through a simple and flexible marking mechanism, thus achieving high statistical multiplexing gain for real-time traffic. By employing constant mean rate reservation on each cycle of the ring and along with a simple but effective CAC function, which admits real-time connections bounding under a predefined quota ratio, the scheme can efficiently accommodate VBR traffic fluctuation. If the value of the quota ratio is set reasonably, the probability of the fluctuated VBR traffic fails to transfer due to expired quota is significantly small, thereby achieving exceedingly low VBR delay and jitter. Additionally, we develop a novel approximation to acquire the accurate results of the expected connection setup delay by means of an M/G/m queueing analysis. In the analysis, the maximum admissible quota of real-time traffic is regarded as the number of servers and the service time has a duration that follows an exponential form with an added constant. Unlike most of the proposed approximation only maintained a less than 10% relative error for certain properties of service distributions, our main contribution is to yet accurate expected waiting time for a multi-server queueing system with the specific service time in our system. Extensive simulation results show that the mean setup queueing time is in profound agreement with the analytic result, and that PQOC/QA achieves remarkable real-time traffic performance while still retaining maximal aggregate system throughput.

References

- [1] B. Mukherjee, "WDM Optical Communication Networks: Progress and Challenges," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, Oct. 2000, pp. 1810-1824.
- [2] P. Green, "Progress in optical networking," *IEEE Communications Magazine*, vol. 39, no. 1, Jan. 2001, pp. 54-61.
- [3] M. Herzog, M. Maier, and M. Reisslein, "Metropolitan Area Packet-Switched WDM Networks: A Survey on Ring Systems," *IEEE Communications Surveys & Tutorials*, vol. 6, no. 2, 2004, pp. 2-20.
- [4] S. Yao, S. Yoo, B. Mukherjee, and S. Dixit, "All-Optical Packet Switching for Metropolitan Area Networks: Opportunities and Challenges," *IEEE Communications Magazine*, vol. 39, no. 3, Mar. 2001, pp. 142-148.
- [5] R. Jain, "Optical Networking: Recent Developments, Issues, and Trends," *Tutorial at IEEE Infocom 2003*, San Francisco, CA, Mar. 2003.
- [6] R. Doverspike, S. Phillips, and J. Westbrook, "Transport Network Architectures in an IP World," *Proc. IEEE Infocom, Tel Aviv, Israel*, Mar. 2000, pp. 305-314.
- [7] M. Yuang, Y. Lin, S. Lee, I. Chao, B. Lo, P. Tien, C. Chien, J. Chen, and C. Wei, "HOPSMAN: An Experimental Testbed System for a 10-Gb/s Optical Packet-Switched WDM Metro Ring Network," *IEEE Communications Magazine*, vol. 46, no. 7, July 2008, pp. 158-166.
- [8] M. Yuang, I. Chao, and B. Lo, "HOPSMAN: An Experimental Optical Packet-Switched Metro WDM Ring Network with High-Performance Medium Access Control," *IEEE/OSA Journal of Optical Communications and Networking*, 2010, vol. 2, no. 2, Feb 2010, pp. 91-101.
- [9] I. White, M. Rogge, K. Shrikhande, and L. Kazovsky, "A Summary of the HORNET Project: A Next-Generation Metropolitan Area Network," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 9, Nov. 2003, pp. 1478-1494.
- [10] A. Carena, V. Feo, J. Finochietto, R. Gaudino, F. Neri, C. Piglione, and P. Poggiolini, "RingO: An Experimental WDM Optical Packet Network for Metro Applications," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 8, Oct. 2004, pp. 1561-1571.
- [11] M. Marsan, A. Bianco, E. Leonardi, A. Morabito, and F. Neri, "All-Optical WDM Multi-Rings with Differentiated QoS," *IEEE Communications Magazine*, vol. 37, no. 2, Feb. 1999, pp. 58-66.
- [12] C. Develder et al., "Benchmarking and Viability Assessment of Optical Packet Switching for Metro Networks," *IEEE Journal of Lightwave Technology*, vol. 22, no. 11, Nov. 2004, pp. 2435-2451.
- [13] C. Linardakis, H. Leligou, A. Stavdas, and J. Angelopoulos, "Implementation of medium access control for interconnecting slotted rings to form a WDM

- metropolitan area network”, *Journal Of Optical Networking*, vol. 3, no. 11, Nov. 2004, pp. 826-836.
- [14] C. Jelger and J. Elmirghani, ”A Slotted MAC Protocol for Efficient Bandwidth Utilization in WDM Metropolitan Access Ring Networks,” *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 8, Oct. 2003, pp. 1295-1305.
- [15] C. Linardakis, H. Leligou, A. Stavdas, and J. Angelopoulos, “Using Explicit Reservations to Arbitrate Access to a Metropolitan System of Slotted Interconnected Rings Combining TDMA and WDMA,” *IEEE Journal of Lightwave Technology*, vol. 23, no. 4, April 2005, pp. 1576–1585.
- [16] J. Kim, J. Cho, S. Das, D. Gutierrez, M. Jain, C. Su, R. Rabbat, T. Hamada, and L. Kazovsky, “Optical Burst Transport: A Technology for the WDM Metro Ring Networks,” *IEEE Journal of Lightwave Technology*, vol. 25, no. 1, Jan. 2007, pp. 93–102.
- [17] L. Lenzini, J. Limb, I. Rubin, W. Lu, and M. Zukerman, “Analysis and synthesis of MAC protocols,” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 9, Sep. 2000, pp. 1557–1562.
- [18] N. Dono, M. Chen, and R. Ramaswami, “A media access control protocol for packet switched wavelength division multiaccess metropolitan networks,” *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 8, Aug. 1990, pp. 1048–1057.
- [19] P. Dowd, K. Borgineni, and K. Sivalingam, “Low-complexity multiple access protocols for wavelength-division multiplexed photonic networks,” *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 4, May 1993, pp. 590–604.
- [20] M. Yuang, Y. Lin, and Y. Wang, “A Novel Optical-Header Processing and Access Control System for a Packet-Switched WDM Metro Ring Network,” *IEEE Journal of Lightwave Technology*, vol. 27, no. 21, Nov. 2009, pp. 4907–4915.
- [21] B. Mukherjee, *Optical Communication Networks*, the University of California, Davis, McGraw-Hill, 1997.
- [22] B. Mukherjee, *Optical WDM Networks*, New York: Springer, 2006.
- [23] J. Cai, A. Fumagalli, and I. Chlamtac, “The Multitoken Interarrival Time (MTIT) Access Protocol for Supporting Variable Size Packets over WDM Ring Network,” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, Oct. 2000, pp. 2094-2104.
- [24] C. Jeiger and J. Elmirghani, “Photonic Packet WDM Ring Networks Architecture and Performance,” *IEEE Communications Magazine*, vol. 40, no. 11, Nov. 2002, pp. 110–115.
- [25] M. Maier, *Metropolitan Area WDM Networks—An AWG Based Approach*, Norwell, MA: Kluwer, 2003.
- [26] K. Imai, T. Ito, H. Kasahara, and N. Morita, “ATMR: asynchronous transfer mode ring protocol,” *Computer Networks and ISDN Systems*, vol. 26, no. 6-8, March 1994, pp. 785-798.

- [27] I. Cidon and Y. Ofek, "MetaRing - A Full-Duplex Ring with Fairness and Spatial Reuse," *IEEE Transactions on Communications*, vol. 41, no. 1, Jan. 1993, pp. 969-981.
- [28] F. Davik, M. Yilmaz, S. Gjessing, and N. Uzun, "IEEE 802.17 Resilient Packet Ring Tutorial," *IEEE Communications Magazine*, vol. 42, no. 3, Mar. 2004, pp. 112-118.
- [29] C. Huang, H. Peng, and F. Yuan, "A Deterministic Bound for the Access Delay of Resilient Packet Rings," *IEEE Communications Letters*, vol. 9, no. 1, Jan 2005, pp. 87-89.
- [30] Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN), *IEEE Standard 802.6*, Dec. 1990.
- [31] M. Marsan, A. Bianco, E. Leonardi, A. Morabito, and F. Neri, "All-Optical WDM Multi-Rings with Differentiated QoS," *IEEE Communications Magazine*, vol. 37, no. 2, Feb. 1999, pp. 58-66.
- [32] K. Bengi, and H. Van As, "Efficient QoS Support in a Slotted Multihop WDM Metro Ring," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 1, Jan. 2002, pp. 216-227.
- [33] H. Leligou, J. Angelopoulos, C. Linardakis, and A. Stavdas, "A MAC protocol for efficient multiplexing QoS-sensitive and best-effort traffic in dynamically configurable WDM rings," *Journal Of Computer Networks*, vol. 44, no. 3, Feb. 2004, pp. 305-317.
- [34] L. Wang, M. Hamdi, R. Manivasakan, and D. Tsang, "Multimedia-MAC protocol: its performance analysis and applications for WDM networks," *IEEE Transactions on Communications*, vol. 54, no. 3, Mar. 2006, pp. 518-531.
- [35] H. Lin, W. Chang, and H. Wu, "FARE: An efficient integrated MAC protocol for differentiated services in WDM metro rings," *Computer Communications*, vol. 30, no. 6, Mar. 2007, pp. 1315-1330.
- [36] A. Fumagalli, J. Cai, and I. Chlamtac, "A Token-Based Protocol for Integrated Packet and Circuit Switching in WDM Rings," in *Proc. IEEE GLOBECOM*, vol. 4, Sydney, Australia, Nov. 1998, pp. 2339-44.
- [37] M. Ma and M. Hamdi, "Providing Deterministic Quality-of-Service Guarantees on WDM Optical Networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, Oct. 2000, pp. 2072-2083.
- [38] P. Sarigiannidis, S. Petridou, G. Papadimitriou, M. Obaidat, and A. Pomportsis, "Supporting Quality-of-Service Scheduling in a TT-FR WDM System," *IEEE SYSTEMS JOURNAL*, vol. 2, no. 4, Dec. 2008, pp. 525-535.
- [39] J. Diao, and PL. Chu, "Packet rescheduling in WDM star networks with real-time service differentiation," *IEEE Journal of Lightwave Technology*, vol. 19, no. 12, Dec. 2001, pp. 1818-1828.
- [40] A. Yan, A. Ganz, and C. M. Krishna, "A Distributed adaptive protocol providing real-time services in WDM-based LANs," *IEEE Journal of Lightwave Technology*, vol. 14, June 1996, pp. 1245-1254.

- [41] I. Akyildiz, J. McNair, L. Martorell, R. Puigjaner, and Y. Yesha, "Medium access control protocols for multimedia traffic in wireless networks," *IEEE Network*, vol. 13, no. 4, Jul./Aug. 1999, pp. 39–47.
- [42] C. Krishna, A. Yan, and A. Ganz, "A distributed adaptive protocol providing real-time services on WDM-based LANs," *IEEE Journal of Lightwave Technology*, vol. 14, no. 6, Jun. 1996, pp. 1245–1254.
- [43] F. Vakil, "A Capacity Allocation Rule for ATM Networks," in *Proc. IEEE GLOBECOM*. 1993.
- [44] S. Chatziperis, P. Koutsakis, and M. Paterakis, "A New Call Admission Control Mechanism for Multimedia Traffic over Next-Generation Wireless Cellular Networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, Jan. 2008, pp. 95-112.
- [45] R. GuCrin, H. Ahmadi, and M. Naghshineh, "Equivalent Capacity and Its Application to Bandwidth Allocation in High-speed Networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 7, Sept. 1991, pp. 968-981.
- [46] H. Perros and K. Elsayed, "Call Admission Control Schemes: A Review," *IEEE Communications Magazine*, vol. 34, no. 11, Nov. 1996, pp. 82-91.
- [47] B. Li, L. Li, B. Li, K. Sivalingam, and X. Cao, "Call Admission Control for Voice/Data Integrated Cellular Networks: Performance Analysis and Comparative Study," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 4, May 2004, pp. 706-718.
- [48] Z. Ali, W. Sheikh, E. Chong, and A. Ghafoor, "A Scalable Call Admission Control Algorithm," *IEEE/ACM Transactions on Networking*, vol. 16, no. 2, Apr. 2008, pp. 424-434.
- [49] M. Mak et al., "Widely Tunable Polarization-Independent All-Optical Wavelength Converter using a Semiconductor Optical Amplifier," *IEEE Photonics Technology Letters*, vol. 12, no. 5, May 2000, pp. 525-527.
- [50] L. Kleinrock, *Queueing Systems*, vols. 1 and 2. John Wiley and Sons, New York, 1975.
- [51] K. Altinkemer, I. Bose, and R. Pal, "Average waiting time of customers in an M/D/k queue with nonpreemptive priorities," *Computers and Operations Research*, vol. 25, no. 4, 1998, pp. 317-328.
- [52] J. Mayhugh and R. McCormick, "Steady State Solution of the Queue M/E_k/r," *Management Science*, vol. 14, no. 11, Jul. 1968, pp. 692-712.
- [53] F. Hillier and F. Lo, "Tables for Multi-Server Queueing Systems Involving Erlang Distribution," *Technical Report 31*, Department of Operations Research, Stanford University, 1971.
- [54] L. Seelen, H. Tijms, and M. Van Hoorn, *Tables for Multiserver Queues*, North-Holland, Amsterdam, 1984.
- [55] F. Barceló and J. Paradells, "The M/H₂/s Queue in Mobile Communications: Approximation of the Mean Waiting Time," *14th U.K. Teletraffic Symposium, IEE*, 1997.

- [56] J. De Smit, "A Numerical Solution for the Multi-Server Queue with Hyper-Exponential Service Times," *Operations Research Letters*, vol. 2, no. 5, 1983, pp. 217.
- [57] E. Maaloe, "Approximation Formulae for Estimation of waiting-time in Multiple-Channel Queueing System," *Management Science*, vol. 19, no. 6, Feb. 1973, pp. 703-710.
- [58] GP. Cosmetatos, "Some Approximate Equilibrium Results for the Multi-Server Queue (M/G/r)," *Operational Research Quarterly*, vol. 27, no. 3, 1976, pp. 615-620.
- [59] Y. Takahashi, "An Approximation Formula for the Mean Waiting Time of an M/G/c Queue," *Journal of the Operations Research Society of Japan*, vol. 20, no. 3, Sept. 1977, pp. 150-163.
- [60] S. Nozaki, and S. Ross, "Approximations in Finite Capacity Multiserver Queues with Poisson Arrivals," *Journal of Applied Probability*, vol. 15, no. 4, Dec, 1978, pp. 826-834.
- [61] F. Kelley, *Reversibility and Stochastic Networks*, Wiley, New York, 1979.
- [62] O. Boxma, J. Cohen, and N. Huffels, "Approximations of the Mean Waiting Time in an M/G/s Queueing System," *Operations Research*, vol. 27, no. 6, Nov. 1979, pp. 1115-1127.
- [63] H. Tijms, M. Van Hoorn, and A. Federgruen, "Approximations for the steady-state probability in the M/G/c queue," *Advances in Applied Probability*, vol. 13, no. 1, Mar. 1981, pp. 186-206.
- [64] T. Kimura, "Diffusion Approximation for an M/G/m Queue," *Operations Research*, vol. 31, no. 2, Mar.-Apr. 1983, pp. 304-321.
- [65] D. Yao, "Refining the Diffusion Approximation for the M/G/m Queue," *Operations Research*, vol. 33, no. 6, Nov.-Dec. 1985, pp. 1266-1277.
- [66] T. Kimura, "Approximations for Multi-Server Queues: System Interpolations," *Queueing Systems*, vol. 17, no. 4, Dec. 1994, pp. 347-382.
- [67] C. Wang, "Light Traffic Approximations for Regenerative Queueing Processes," *Advances in Applied Probability*, vol. 29, no. 4, Dec. 1997, pp. 1060-1080.
- [68] C. Wang and R. Wolff, "The M/G/c queue in light traffic," *Queueing Systems*, vol. 29, no. 1, Aug. 1998, pp. 17-34.
- [69] C. Wang and R. Wolff, "Systems with Multiple Servers under Heavy-tailed Workloads," *Performance Evaluation*, vol. 62, no. 4, Aug. 2005, pp. 456-474.
- [70] W. Fischer and K. Meier-Hellstern, "The Markov-modulated Poisson Process (MMPP) Cookbook," *Performance Evaluation*, vol. 18, no. 2, Sept. 1993, pp. 149-171.
- [71] P. Tang et al., "Rapidly Tunable Optical Add-Drop Multiplexer (OADM) using a Static-Strain-Induced Grating in LiNbO₃," *IEEE Journal of Lightwave Technology*, vol. 21, Jan. 2003, pp. 236-245.
- [72] S. Huang et al., "Experimental Demonstration of Active Equalization and ASE Suppression of Three 2.5-Gb/s WDM-Network Channels over 2500 km using

- AOTF as Transmission Filters,” *IEEE Photonics Technology Letters*, vol. 9, Mar. 1997, pp. 389-391.
- [73] A. Sneh et al., "High-Speed Wavelength Tunable Liquid Crystal Filter," *IEEE Photonics Technology Letters*, vol. 7, Apr. 1995, pp. 379-381.
- [74] R. Cooper and S. Niu, “Benes's Formula for M/G/1-FIFO 'Explained' by Preemptive-Resume LIFO,” *Journal of Applied Probability*, vol. 23, no. 2, Jun., 1986, pp. 550-554.
- [75] F. Kelly, “The Departure Process from a Queueing System,” *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 80, no. 2, Sept. 1976, pp. 283-285.
- [76] R. Cooper, *Introduction to Queueing Theory*, the George Washington University, Washington, D. C.: CEEPress, 1990.



Vita



I-Fen Chao received B.S. and M.S. degrees in computer and information engineering from National Chiao Tung University, Taiwan, in 1992 and 1994, respectively. From 1995 to 1998 she was at CCL/ITRI, working on personal communications systems. From 1998 to 2003 she was with Faraday Technology Corporation, Hsinchu Science Park, Taiwan, as a technical manager working on an embedded OS/system. In 2003, she joined Computer and Information Engineering, National Chiao Tung University, where she is currently pursuing A Ph.D. degree. Her current research interests include high-speed networking, optical networking, and performance modeling and analysis.

住址：新竹市東區大學路 68 號 4F-2