NOTE TO USERS

This reproduction is the best copy available.

UMI[®]

Fair Medium Access Control Scheme for Packet Ring Networks

by
Yiqun Helen Tang

A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the
Ottawa-Carleton Institute for Electrical and Computer Engineering
Department of Systems and Computer Engineering
Carleton University

Ottawa, Ontario, Canada K1S 5B6

© Copyright by Yiqun Helen Tang, 2005

Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-494-13400-3

Our file Notre référence

ISBN: 0-494-13400-3

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

ABSTRACT

In recent years, the rapidly increasing volume of data traffic in Metropolitan Area Networks (MANs) is challenging the capacity limits of existing transport infrastructures based on circuit-oriented technologies such as Synchronous Optical Network (SONET). Packet ring networks have attracted wide attention as the transport technology for data services in MAN. It combines the resilience and deterministic performance of SONET and the data friendliness of Ethernet switches. However, since packet rings allow concurrent transmission of packets over different segments, the ring bandwidth becomes a shared medium. Hence, a key challenge for packet rings is to design a Medium Access Control (MAC) scheme that ensures all nodes have fair access to the ring.

In this study, we investigate the existing MAC schemes including the IEEE RPR standard and its enhancements, in which ring access is provided through fair rate estimation and feedback control. The existing approaches suffer from some performance limitations such as large access delay and throughput loss.

The deficiencies of the existing schemes motivate the research of this thesis: fair MAC scheme for packet rings. Our solution is the design of a novel MAC scheme, called Virtual Source Queuing (VSQ). VSQ offers guaranteed fair medium access by providing individual virtual queues and fair queuing to all the competing sources. We also propose a simple flow control scheme called One-Hop-Backpressure (OHB) to provide lossless transit paths for VSQ. OHB exploits the source information provided by VSQ to effectively control the transit buffer occupancy through implicitly controlling the input

rate of the traffic sources. As a result, VSQ requires less transit buffer size to achieve a lossless transit path than RPR does.

We conduct in-depth performance evaluations of VSQ and RPR. Our analysis and simulation results reveal that VSQ not only meets all the MAC design goals for packet rings, such as guaranteed fairness and maximum throughput, but also offers faster fairness convergence, less access delay and less transit buffer occupancy in comparison to RPR.

DEDICATION

This thesis is dedicated to the memory of my mother

Qinmei Yu

(1950-2005)

ACKNOWLEDGEMENT

First, I would like to sincerely thank my thesis supervisor Professor Ioannis Lambadaris for his help and effort during the course of this work. His invaluable guidance and knowledgeable suggestions have greatly enhanced my work. I would like to thank Professor Jerome Talim. His advice and feedback about my research have strengthened the work. I would like to also thank other members of my committee for the assistance and valuable input they provided at all levels of my graduate studies.

I greatly appreciate the generous financial support for my studies provided by the Ontario Graduate Scholarship (OGS), Communications and Information Technology Ontario (CITO), Mathematics of Information Technology and Complex Systems (MITACS), Roger Kaye Memorial Scholarships and Carleton University.

My deep gratitude is due to my family and friends for their encouragement and support, which was crucial in motivating me to complete this thesis. In particular, I would like to thank my father, Zhengzhong Tang, my late mother Qinmei Yu, my mother-in-law, Habibeh Kia, my husband, Hamid Mehrvar, My brother Yihui Tang, for their constant love, support and encouragement, and my beloved daughter, Parisa Linna Mehrvar, for her enjoyable company during the process of my thesis preparation. I would also like to thank my friends at Broadband Networks Lab for providing a friendly working environment and helpful discussions.

TABLE OF CONTENTS

ABSTRACT	II
DEDICATION	IV
ACKNOWLEDGEMENT	V
TABLE OF CONTENTS.....	VI
LIST OF FIGURES	VIII
LIST OF TABLES	IX
LIST OF ACRONYMS.....	X
CHAPTER 1 INTRODUCTION	1
1.1 TECHNOLOGIES FOR DATA SERVICES IN MAN.....	1
1.1.1 <i>SONET</i>	1
1.1.2 <i>Ethernet</i>	2
1.1.3 <i>Packet Ring Networks</i>	3
1.2 MAC SCHEMES FOR PACKET RINGS	5
1.2.1 <i>Design Goals</i>	5
1.2.2 <i>Existing MAC schemes and their limitations</i>	6
1.2.3 <i>Towards a New MAC Scheme: Virtual Source Queuing</i>	8
1.3 THESIS CONTRIBUTIONS.....	9
1.4 THESIS ORGANIZATION	11
CHAPTER 2 MAC SCHEMES FOR PACKET RING NETWORKS	12
2.1 INTRODUCTION.....	12
2.2 THE RPR STANDARD	12
2.2.1 <i>RPR MAC Data Path Sub-layer</i>	14
2.2.2 <i>RPR MAC Control Sub-layer</i>	19
2.2.3 <i>Starvation Problem and Fairness Control</i>	24
2.2.4 <i>RPR Node Architecture</i>	26
2.2.5 <i>Performance Limitations of RPR MAC</i>	28
2.3 ENHANCEMENTS TO RPR FAIRNESS ALGORITHM	33
2.4 SUMMARY	44
CHAPTER 3 DESCRIPTION OF VIRTUAL SOURCE QUEUING (VSQ)	46
3.1 INTRODUCTION.....	46
3.2 RESEARCH PROBLEM	46
3.3 VSQ MAC SCHEME	50
3.3.1 <i>Description</i>	51
3.3.2 <i>VSQ Node Architecture</i>	54
3.4 FLOW CONTROL SCHEME FOR VSQ: OHB.....	57
3.4.1 <i>Description</i>	57
3.4.2 <i>Analysis</i>	63
3.5 PROPERTIES OF VSQ.....	66
3.5.1 <i>Fair Rate Guarantee</i>	67
3.5.2 <i>Fairness Bound</i>	68
3.5.3 <i>Access Delay Bound</i>	70
3.5.4 <i>Isolation</i>	71

3.6	RECOMMENDATIONS FOR SETTING VSQ PARAMETERS	71
3.6.1	<i>Weight</i>	72
3.6.2	<i>Buffer Thresholds</i>	74
3.7	SUMMARY	76
CHAPTER 4 PERFORMANCE EVALUATION OF VSQ		80
4.1	INTRODUCTION	80
4.2	EVALUATION METHODOLOGY	80
4.2.1	<i>Analytical Modeling</i>	80
4.2.2	<i>Simulation Tool</i>	81
4.2.3	<i>Performance Metrics</i>	81
4.2.4	<i>Simulation Setup</i>	82
4.2.5	<i>Format of the Results</i>	84
4.3	PROPERTIES OF VSQ	84
4.3.1	<i>Weighted Fairness</i>	84
4.3.2	<i>Spatial Reuse</i>	86
4.3.3	<i>Dynamic Responsiveness</i>	88
4.3.4	<i>Isolation of Ill-behaving Sources</i>	89
4.4	THROUGHPUT COMPARISON	91
4.5	FAIRNESS CONVERGENCE TIME COMPARISON	92
4.6	ACCESS DELAY BOUND ANALYSIS	95
4.6.1	<i>Access Delay Bound for VSQ</i>	96
4.6.2	<i>Access Delay Bound for RPR</i>	97
4.6.3	<i>Comparison</i>	103
4.7	AVERAGE DELAY ANALYSIS	105
4.7.1	<i>System Assumptions</i>	106
4.7.2	<i>Average Delay Analysis for VSQ</i>	107
4.7.3	<i>Average Delay Analysis for RPR</i>	110
4.7.4	<i>Comparison Using Poisson Model</i>	114
4.7.5	<i>Comparison Using Bursty Model</i>	115
4.8	TRANSIT BUFFER REQUIREMENT	118
4.8.1	<i>Analysis</i>	118
4.8.2	<i>Simulation Results on the Transit Buffer Occupancy</i>	122
4.8.3	<i>Simulation Results on the Transit Buffer Occupancy Variance</i>	126
4.9	SUMMARY	127
CHAPTER 5 CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH		130
5.1	THESIS SUMMARY	130
5.2	MAC DESIGN GOALS RE-EXAMINED	131
5.3	VSQ CLARIFICATIONS	135
5.4	CONTRIBUTIONS	136
5.5	RECOMMENDATIONS FOR FUTURE RESEARCH	137
BIBLIOGRAPHY		139

LIST OF FIGURES

Figure 1-1: The concept of spatial reuse.....	4
Figure 2-1: IEEE 802 / RPR layer as data link layer of OSI reference model.....	13
Figure 2-2: Illustration of BIR.....	14
Figure 2-3: RPR MAC with 2-TB	17
Figure 2-4: Important functions of MAC Control sub-layer	19
Figure 2-5: Normal operation versus Wrapped state.....	22
Figure 2-6: Illustration of RPR Fairness Algorithm	24
Figure 2-7: RPR node architecture from fairness algorithm viewpoint.....	26
Figure 2-8: Unbalanced traffic demands	29
Figure 2-9: Simulation results demonstrating bandwidth oscillations with RPR-AM	30
Figure 2-10: Illustration of remote fair queuing: (a) GPS server (b) DVSR approximation	34
Figure 2-11: Measurements of the arrival rate (A_i)	34
Figure 2-12: DVSR fair rate calculation.....	35
Figure 2-13: Illustration for Example 2.2.....	40
Figure 3-1: Illustration of the research problem.....	49
Figure 3-2: Illustration of VSQ	52
Figure 3-3: Illustration of the traffic flow isolation with VSQ.....	53
Figure 3-4: Pseudo code for VSQ	55
Figure 3-5: VSQ node architecture.....	56
Figure 3-6: Illustration of OHB scheme	58
Figure 3-7: Pseudo code of OHB scheme	59
Figure 3-8: Illustration of VSQ transit buffer usage.....	74
Figure 3-9: MAC layer comparison of VSQ, RPR and OSI reference model	78
Figure 4-1: Weighted fairness setup.....	85
Figure 4-2: Weighted fairness results.....	85
Figure 4-3: Scenario with two bottleneck links: link 3-4 and link 7-8.	87
Figure 4-4: Throughput of 6 flows from two congestion domains.....	87
Figure 4-5: Packing lot scenario with dynamic traffic	88
Figure 4-6: Results showing fast reclamation of available bandwidth.	88
Figure 4-7: Scenario to show isolation.....	90
Figure 4-8: Results showing isolation property.....	90
Figure 4-9: Unbalanced traffic scenario with two flows	92
Figure 4-10: Link utilization comparison of RPR and VSQ	92
Figure 4-11: Scenario to study convergence time	93
Figure 4-12: Convergence of RPR	94
Figure 4-13: Convergence of VSQ.....	94
Figure 4-14: Fair rate convergence time comparison	95
Figure 4-15: Illustration of access delay bound derivation in VSQ.....	97
Figure 4-16: Time diagram for fair rate control cycle: $responseTime \sim (t_6 - t_0)$	98
Figure 4-17: Illustration of a congestion domain and burst concatenation.....	101
Figure 4-18: Illustration of $maxFRED$ derivation for RPR-2TB.....	102
Figure 4-19: Access delay comparison of RPR and VSQ	114
Figure 4-20: End-to-end delay comparison of RPR and VSQ.....	115
Figure 4-21: ON/OFF Pareto traffic model	116
Figure 4-22: Access delay comparison using Pareto traffic model	117
Figure 4-23: E2E delay comparison using Pareto traffic model.....	118
Figure 4-24: Transit buffer size requirement comparison for Example 4-1	122
Figure 4-25: Simulation setup to study buffer occupancy	123
Figure 4-26: Transit buffer occupancy comparison: VSQ vs RPR.....	124
Figure 4-27: Transit buffer average and variance comparison: VSQ vs. RPR.....	126

LIST OF TABLES

Table 2-1: Service classes and their QoS relationship as defined in RPR standard	16
Table 3-1: Design goals compliance comparison	48
Table 3-2 Comparison of feedback control for RPR and VSQ	61
Table 3-3: Comparison of flow control schemes	63
Table 4-1: Characteristics of the test network	82
Table 4-2: RPR parameters	82
Table 4-3: Simulation Scenarios and Results	83
Table 5-1: Relative performance comparison of different MAC schemes	134
Table 5-2: Complexity comparison between different MAC schemes	136

LIST OF ACRONYMS

AM	Aggressive Mode
ATM	Asynchronous Transfer Mode
BIR	Buffer Insertion Ring
CBFQ	Credit-Based Fair Queuing
CBR	Constant Bit Rate
CIR	Committed Information Rate
CM	Conservative Mode
DoS	Data over SONET
DRR	Deficit Round Robin
DS	Digital Signal
DVSR	Distributed Virtual time Scheduling in Rings
EIR	Excess Information Rate
FB	Fairness Bound
FCFS	First Come First Serve
FE	Fairness Eligible
FIFO	First In First Out
FRTT	Fairness Round Trip Time
GigE	Gigabit Ethernet
GFP	Generic Framing Procedure
GPS	Generalized Processing Sharing
HBH	Hop-by-Hop

HOL	Head-of-Line
HP	High Priority
IB	Ingress Buffer
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
ITU	International Telecommunication Union
LAN	Local Area Networks
LCAS	Link Capacity Adjustment Scheme
LCDBA	Low Complexity Distributed Bandwidth Allocation
LP	Low Priority
MAC	Medium Access Control
MAN	Metropolitan Area Networks
MaxFRED	Maximum Fair Rate Enforcement Delay
MaxFRTT	Maximum Fairness Round Trip Time
OAM	Operation, Administration, Maintenance
OC	Optical Carrier
OHB	One-Hop-Backpressure
OTN	Optical Transport Network
PCL	Pseudo-Conservation Law
PHY	physical layer
POS	Packet-Over-SONET
QoS	Quality of Service
RIAS	Ring Ingress Aggregated with Spatial Reuse

RPR	Resilient Packet Ring
RTT	Round Trip Time
SCFQ	Self-Clocked Fair Queuing
SCI	Scalable Coherent Interface
SFQ	Stochastic Fair Queuing
SLA	Service Level Agreement
SONET	Synchronous Optical Network
TB	Transit Buffer
TDM	Time Division Multiplexing
VDQ	Virtual Destination Queuing
VSQ	Virtual Source Queuing
VQ	Virtual Queuing
WFQ	Weighted Fair Queuing
WRR	Weighted Round Robin

CHAPTER 1 Introduction

An important trend in networking is the migration of packet-based technologies from Local Area Networks (LANs) to Metropolitan Area Networks (MANs)¹. The rapidly increasing volume of data traffic in MANs is challenging the capacity limits of the existing transport infrastructures that are based on circuit-oriented technologies such as Synchronous Optical Network (SONET) and Asynchronous Transfer Mode (ATM). Inefficiencies associated with carrying increasing quantities of data traffic over voice-optimized circuit-switched networks make it difficult to provision new services. On the other hand, the cost of building additional capacity is beyond the limits of most carriers' capital expense budgets. Packet-based transport technology is considered by many to be the only alternative for scaling metropolitan networks to meet the demand [1]. Finding the optimal transport technology for data services in MAN has been an active area of research over the past few years.

1.1 Technologies for data services in MAN

In this section, we describe three main technologies available to support data services in a MAN: SONET, Ethernet and Packet Rings. Because packet rings combine the advantages of SONET and Ethernet, it is the focus of this thesis.

1.1.1 SONET

Most optical fiber in the metro area is in ring form because of the form's simplicity and robustness. SONET is optimized to transport voice traffic by provisioned point-to-

¹ Term Metro Networks or simply 'metro' have been commonly used instead of Metropolitan Area Networks (MANs).

point links and offers guaranteed Quality-of-Service (QoS) performance in terms of jitter and latency [2]. SONET is a lossless transport in which packets are delivered from source to destination without any QoS decisions in the intermediate nodes. In addition, SONET is known for providing high availability by using spare fibers or additional capacity to provide protection in case of fiber cut or equipment failure. It is known that SONET can provide 50 *msec* protection switching in a ring of 16 nodes and 1200 *km* distance [3]. To transport data traffic, improvements have been implemented. The technologies behind the next-generation SONET are commonly known as the Data over SONET (DoS) concept. The DoS scheme consists of three technologies: Generic Framing Procedure (GFP) [4], Virtual Concatenation (VCAT) [5] and Link Capacity Adjustment Scheme (LCAS) [6], all standardized by ITU-T and Standards Committee. Nonetheless, since SONET was primarily designed for point-to-point, circuit-switched applications such as voice traffic, it is not the most efficient transport for bursty data traffic. In addition, bandwidth is reserved for every source on the ring, meaning no statistical multiplexing gain could be obtained.

1.1.2 Ethernet

Unlike SONET, Ethernet makes efficient use of available bandwidth for data traffic. It is a simple and inexpensive solution for data traffic and it has become the de facto standard for data networking within Local Area Networks. Ethernet switch vendors have leveraged recent advances in Ethernet, including the development of Gigabit Ethernet [7] and long-reach Ethernet components [8][9][10], to deploy switched Ethernet solutions in MANs. Although Ethernet switch solutions have the appeal of being low cost and data-oriented, these solutions are neither optimized for the ring topology that dominates metro

networks, nor do they support the levels of survivability or service-level guarantees expected in a public-network infrastructure. Recovery from network faults could take tens of seconds versus the 50 *msec* performance of SONET networks. Additionally, Ethernet switches lack the ability to support deterministic and verifiable service-level agreements. As frames get switched one node at a time, there are no mechanisms to guarantee delay and jitter across multiple switches configured in either a ring or mesh topology. Ethernet switches are not the answer to building scalable metropolitan networks that can deliver high availability and QoS guarantee.

1.1.3 Packet Ring Networks

The use of pure SONET for data services is inefficient while using pure Ethernet does not guarantee QoS. As a result, there is a compelling need for a new architecture for metropolitan packet networks: an architecture that combines the resilience and deterministic performance of SONET and the data friendliness of Ethernet switches. In the recent years, researchers started to look at spatial reuse packet rings [11][12] as an alternative solution to provide data services in MANs. Spatial reuse packet rings combine the best of both SONET and Ethernet technologies.

To describe the concept of *spatial reuse*, let us first consider the well-known token ring. In a token ring, a packet originates from a source and travels across the ring, is copied by its destination and then removed by the source. On the other hand, in a spatial reuse ring, a packet is removed by its destination so that the packet does not need to travel through the remaining nodes. This makes concurrent transmission over distinct segments of the ring possible. As a result, the total effective ring throughput can be

significantly higher than the capacity of a single link. If the traffic on the ring is uniformly distributed between all source/destination pairs, then the throughput achieved with spatial reuse is twice the throughput of the same ring without spatial reuse. This is due to the fact that each packet travels only half the ring on average. In addition, since spatial reuse ring can utilize the bandwidth traditionally set aside for SONET protection and strips data from the ring when it reaches its destination, leaving spare bandwidth to be reused. This results a fourfold increase of the effective bandwidth over the equivalent SONET ring [11]. Figure 1-1 shows a bi-directional ring with 8 nodes connected in tandem via ring interfaces. Spatial reuse is illustrated on the inner ring. Node 8 and node 4 can transmit at the same time as their destination does not overlap with each other.

Figure 1-1. The concept of spatial reuse

The recognition of the advantage of spatial reuse has generated numerous proposals both for new LAN/MAN architectures and for upgrades of existing standards to support this feature [13][14]. The Resilient Packet Ring (RPR) technology, which is standardized

in IEEE 802.17 [15], is based on the spatial reuse packet ring architecture. The RPR standard is the latest development in a series of ring based network protocols standardized by IEEE. IEEE 802.5 Token Ring [16] and IEEE 1596 Scalable Coherent Interface (SCI) [17] [18] are examples of other ring based IEEE standards. Packet ring based data networks were pioneered by the Cambridge Ring [19] followed by other important network architectures, notably MetaRing [11] FDDI [20], ATMR [21] and CRMA-II [22].

However, since the ring bandwidth becomes a shared medium with spatial reuse, a key challenge for packet rings is to design a Medium Access Control (MAC) scheme that ensures all nodes have fair access to the ring [11] [15]. The MAC schemes play a key role in the performance of ring networks.

1.2 MAC Schemes for Packet Rings

1.2.1 Design Goals

In order to design a good MAC scheme for packet rings, it is essential to first understand the parameters that are important to packet rings. While there are many ways to evaluate a MAC scheme, we consider the following metrics:

- **Fairness:** Fair bandwidth allocation to the source nodes that transmit over a congested link. There are many fairness definitions proposed in the literature. We use the one defined in the RPR standard [15]: *The property that for any given link on the ring, every source receives an equal proportion of fairness eligible capacity. If all sources have equal weights, then all sources have substantially equal access to the available capacity of all links.* This means that if the link bandwidth is C and there

are N sources sharing this link, the fair bandwidth allocation to the sources, i.e., the *fair rate*, is C/N .

- **Fast fairness convergence:** The time it takes for the sources to converge to the fair rate should be short.
- **Efficiency:** Maximizing ring bandwidth utilization and minimizing overhead.
- **Low access delay for node traffic:** access delay is defined as the delay from the time a packet arrives at the node till the time it gets on the ring.
- **Scalability:** Scaleable to a large number of stations
- **Ability to deal with ill-behaving sources:** Ill-behaving nodes should not affect the performance of well-behaving nodes.
- **Simplicity:** The scheme should be simple relative to the processing overhead, the required traffic measurements and hardware requirements.

1.2.2 Existing MAC schemes and their limitations

As mentioned in the previous section, a MAC scheme is needed for packet rings because the ring bandwidth is a shared medium. The RPR MAC is based on the Buffer Insertion Ring (BIR) [23][24] protocol. The basic operation of BIR is as follows. There is an insertion buffer at every node interface to solve the conflict between the data already flowing on the ring and data ready to be transmitted by a node. Non-preemptive priority is given to the ring traffic. When a node wants to access the ring, it only needs to check if the insertion buffer is empty or not. The advantage of BIR scheme is its simplicity since the node interface only needs to consider two classes of traffic: ring traffic and node traffic. However, the main problem with BIR is that it does not provide adequate

protection from a sudden bursty source sending packets at a rate higher than its fair rate. Such a bursty source can significantly increase the ring access delay of the packets from other sources. This phenomenon is referred as *starvation* [24]. We quote from [25]: “The register insertion ring² is capable of high throughputs and has only a slightly greater delay for light loading than the token ring. Its greatest disadvantage is that it loses the fair allocation and guaranteed access provided by the token ring’s round-robin packet service”. This inherent unfairness problem for BIR can be solved by fairness control mechanisms that regulate the access to the ring bandwidth.

The existing fairness control mechanisms for BIR can be classified into two categories: preventive approaches and reactive approaches. Preventive approaches regulate the access opportunities of every node by allocating transmission quotas. Each node is guaranteed transmission of its quota within a specified interval. While these approaches, e.g., MetaRing [11], ensure fair allocation, they lead to throughput degradation as the unused quota of a node cannot be shared by others. On the other hand, with reactive approaches [12], e.g., Feedback to suspend [26], when a node suffers starvation, it triggers backpressure signals to enforce other nodes either to stop their transmission or reduce their rate. Reactive control allows much more flexibility in the allocation of resources. Since users are typically not guaranteed resources by the network, resources can be statistically multiplexed. However, there is always a reaction delay and a chance that correlated traffic bursts will overload the network, causing performance degradation, and hence, congestion.

² Register insertion ring is the same as buffer insertion ring.

The most recent effort in the reactive fairness control category is the RPR standard that defines a fairness algorithm as part of its MAC scheme. The RPR fairness algorithm is based on fair rate estimation and feedback control and works as follows: Once starvation is detected at a node, a fair rate is estimated by this node and sent to all the upstream nodes. The upstream nodes then throttle their traffic to the received fair rate. After a convergence period, the starvation clears and the fair rate increases until next starvation happens.

The RPR fairness algorithm, however, has two main shortcomings. The first is bandwidth oscillation due to inaccuracy in fair rate estimation, which results in throughput loss. The second is a large access delay for the node traffic as a natural result of the delay in the feedback rate control.

To resolve the oscillation problem of RPR, several schemes have been proposed in the literature. For example, Distributed Virtual-time Scheduling in Rings (DVSR) has been recently proposed [27]. DVSR estimates the fair rate by approximating fair queuing using per-ingress counters to measure packet arrival rates. Since DVSR offers some improvements in the accuracy of fair rate estimation at the expense of complexity, the bandwidth oscillation is reduced in comparison to RPR. However, DVSR still suffers from large access delay as RPR does since it also relies on rate feedback control to achieve fairness.

1.2.3 Towards a New MAC Scheme: Virtual Source Queuing

The deficiencies of the existing MAC schemes motivate the research goal of this thesis: a fair MAC scheme for packet rings. Our solution to this problem is the design of a new MAC scheme, called Virtual Source Queuing (VSQ).

VSQ tackles the unfairness issue of BIR from the root cause. It provides guaranteed fair medium access by providing virtual queues and fair queuing to all the competing sources³. In other words, VSQ provides isolation among the competing sources and, thus, results in fair sharing of the ring bandwidth.

We also design an effective flow control scheme called One-Hop-Backpressure (OHB) to ensure a lossless transit path for VSQ. OHB exploits the source information provided by VSQ and implicitly control the input rate of the source nodes. This scheme minimizes the latency for feedback control and results in less transit buffer requirement for a lossless transit path.

We perform in-depth performance analysis and comparisons between VSQ and RPR. This is done using both analytical approaches and extensive simulations. Our results demonstrate that VSQ can achieve both fairness and maximum utilization, while offering fast convergence and less access delay. Last but not least, the extra implementation complexity incurred in buffer management is marginal compared to RPR MAC, which requires a complex signalling scheme.

1.3 Thesis Contributions

We summarize the important contributions of this thesis as follows.

1. Investigating the pros and cons of the existing MAC schemes for packet ring networks. We identified that the overlay fairness control, which is based on “fair rate estimation and feedback”, results in two problems: bandwidth oscillations and large access delay.

³ We define a source as a ring node, not as an end-user.

-
2. Proposing VSQ as a new fair MAC for packet ring networks: The main advantages of the proposed scheme are its simplicity, fair access guarantee, elimination of bandwidth oscillation and reduced access delay.
 3. Proposing a new flow control scheme called OHB to ensure lossless transit paths for VSQ. The main advantage of OHB is its simplicity and effectiveness.
 4. Analysing the performance of the VSQ scheme:
 - a) Implementing simulation models for VSQ;
 - b) Analysing and demonstrating the fairness guarantee and fair rate convergence time for VSQ.
 - c) Providing a closed form analytical solution on access delay and end-to-end delay for VSQ using conventional traffic models.
 - d) Validating numerical results obtained from c) with simulations.
 - e) Providing delay performance for non-conventional traffic models through simulations.
 - f) Analysing the transit buffer requirements of VSQ.
 - g) Providing simulation results of transit buffer occupancy average and variance for VSQ.
 5. Comparing the performance of VSQ and RPR in the following aspects:
 - a) Fairness guarantee and fair rate convergence time.
 - b) Average access delay and end-to-end delay with conventional traffic models and bursty traffic models.
 - c) Transit buffer requirements
 - d) Transit buffer occupancy average and variance for both schemes.

1.4 Thesis Organization

The thesis is organized as follows. Following this introduction, we describe the state-of-the-art MAC schemes for packet ring networks including the IEEE RPR standard and its enhancements in Chapter 2. In Chapter 3, we first describe the research problem. We then propose VSQ as our strategy to provide fair medium access control for packet ring networks. Finally, we discuss the architecture, properties, operation and implementation of VSQ and present the OHB flow control scheme to provide lossless transit paths for VSQ. In Chapter 4, performance results are presented to demonstrate the superiority of the VSQ MAC compared to the RPR MAC. Finally, we summarize the contributions and outline possible directions for future research in Chapter 5.

CHAPTER 2 MAC Schemes for Packet Ring Networks

2.1 *Introduction*

In this chapter, we describe the state-of-the-art MAC schemes for packet ring networks. We first introduce the IEEE RPR standard and its performance limitations. We then describe some of the enhancements to the RPR standard that are available in the literature.

2.2 *The RPR Standard*

A basic knowledge of the IEEE 802.17 RPR standard is needed to fully appreciate the discussion of MAC schemes which is the main topic of this thesis. The RPR standard consists of over 600 pages [15]. There are a few tutorials on the RPR standards in the literatures such as [28][29]. This section is not intended to be a full description of all RPR functionality, rather a brief overview of the important RPR characteristics.

The RPR working group was formed in early 2000 to develop a standard for bi-directional metro packet rings. The RPR standard was approved in June 2004. The key performance objective of the RPR standard is to simultaneously achieve high bandwidth utilization, spatial reuse and fairness. RPR supports up to 255 nodes and it is optimized for rings with a maximum circumference of 2000 kilometers. A typical RPR ring may cover a major city, while the practical size limit of a ring is about one US state. To cover larger areas one will have to interconnect several RPR-rings or connect them using IP-networks.

The RPR standard falls in the category of other well-known link layer standards, such as 802.5 (Token Ring) [16], 802.11 (Wireless LAN) [30] and 802.3 (Ethernet) [31]. It implements a MAC protocol, for access to the shared ring communication medium, which has a client interface similar to that of Ethernet's. Figure 2-1 shows the placement of 802 protocols in the link layer of OSI model [32]. The MAC layer defined by RPR can be split into a data path and a control sub-layer. The details of these two sub-layers are provided in the following subsections. RPR also defines a service interface to the above logical link control sub-layer, and the underlying physical layer. RPR itself does not define a physical layer. Instead, it is designed to run over existing layer one technologies such as Ethernet physical layer, or standard SONET framing layer. This gives vendors and carrier customers the flexibility to choose RPR implementations to meet different networking applications.

Figure 2-1: IEEE 802 / RPR layer as data link layer of OSI reference model

2.2.1 RPR MAC Data Path Sub-layer

Figure 2-2 shows a simplified model of the RPR MAC data path which is based on Buffer Insertion Ring (BIR) [24]. An RPR node contains a separate MAC data path for each of the ringlets. Only the traffic of one of the ringlets is shown in Figure 2-2. Each data path contains logic to strip frames from the ring, store the frames in the transit (insertion) buffer, and select the next frame to be transmitted. The conflict between data ready to be transmitted by a node and the data already flowing on the ring is resolved by the transit buffer on the receiving side of each link. A node may add traffic to the ring only if the transit buffer is empty. The transit buffer is needed to store frames that cannot be passed on to the downstream neighbor immediately, when the node is already transmitting on the outbound link. The RPR standard does not specify the size of this buffer, but it needs to be at least as big as the RPR maximum transfer unit.

Figure 2-2: Illustration of BIR

2.2.1.1 Frame Classes

To support a range of performance requirements, RPR defines three different service classes for frames transmitted: class A, B and C, with strict priority between them. The MAC client has a separate transmission buffer for each traffic class.

- The class A service is the highest priority class. This service class is subdivided into class A0 and class A1. The difference between these two sub-classes is that A0 uses reserved bandwidth that cannot be re-used by the other traffic classes. This class is used by RPR control frames as well as applications with strict demands on latency and jitter, such as interactive audio or video applications.
- The class B service offers a bounded delay on a part of the traffic, which is called Committed Information Rate (CIR) and is not subject to rate limiting by the fairness algorithm which will be described in Section 2.2.5.5. The other part of class B traffic is termed Excess Information Rate (EIR) and is rate limited by fairness algorithm.
- Class C traffic has low priority. This class supports best-effort traffic as it has no guarantees on delay, and is rate limited by the fairness algorithm.

Table 2-1 shows this classification in conjunction with the required QoS. As seen, the two traffic classes C and B-EIR are called *Fairness Eligible (FE)* as the fairness algorithm controls such traffic. In order to fulfill the service guarantees for class A0, A1, and B-CIR traffic, bandwidth needed for these traffic classes is pre-allocated. Bandwidth pre-allocated for class A0 traffic is called *reserved* and can only be utilized by the station holding the reservation. Bandwidth pre-allocated for class A1 and B-CIR traffic is called

reclaimable. Both unallocated and unused reclaimable bandwidth can be used to send FE traffic.

Table 2-1: Service classes and their QoS relationship as defined in RPR standard

Class of Service			Fairness Eligible
Class	Usage Example	Sub-Class	
Class A	Real-time	AA	NO
		AA1	NO
Class B	Near Real-time	B-CTR	NO
		B-EN	YES
Class C	Best Effort	C	YES

2.2.1.2 Frame Reception

A node on an RPR ring must inspect every frame received from its upstream neighbor, and make a decision on how to treat it. If the frame is destined to this node⁴, the frame is copied to the MAC control sub-layer. From here, data frames are passed on to the MAC client, based on the frame type field. The node then decrements the *ttl* field of the frame, and discard the frame if this field is zero. Frames with other destination addresses are put in a transit buffer, waiting to be passed on to the downstream node. The transit buffer (sometimes referred as transit queue) acts as a FIFO queue.

The RPR standard defines two implementations for *Transit Buffers (TB)*: 1-TB and 2-TB . The 1-TB implementation uses one transit buffer for traffic of all three classes. In

⁴ The frames destined to this node could be unicast, multicast or broadcast.

a 2-TB design the primary transit buffer is reserved for class *A* traffic, while both class *B* and *C* use the secondary transit buffer. For simplicity of explanation, we map three traffic classes into two priorities: High Priority (HP) for class *A* traffic and Low Priority (LP) for class *B* and *C* traffic.

Figure 2-3: RPR MAC with 2-TB

The scheduler serves the IBs and the TBs based on strict priority. In 1-TB design, the priority order is: TB, HP IB, and LP IB. For 2-TB implementation, the priority order is: HP TB, HP IB, LP TB and LP IB. Figure 2-3 illustrates the RPR MAC with 2-TB design. The numbers in the circles indicate the crude priority order. The advantage of the 1-TB design is its hardware simplicity. However, it is possible that LP ring traffic blocks HP node traffic from accessing the ring. This is because once the LP traffic from upstream nodes gets on the ring, it becomes ring traffic and assumes higher priority than the HP traffic of a downstream node. On the other hand, the 2-TB design offers relatively lower access delay for HP node traffic that has higher priority than LP ring traffic. The disadvantage, however, is that the LP transit buffer must be large enough to store the LP

ring traffic while yielding the ring access to HP node traffic. In both 1-TB and 2-TB designs, LP ring traffic has priority over LP node traffic. Therefore, a heavily loaded upstream node can hog the ring bandwidth for a very long time and cause starvation for its downstream nodes.

2.2.1.3 Frame Removal

Due to the spatial reuse property of RPR, the destination node is responsible for stripping a frame from the ring. Multicast and broadcast frames are exceptions from this rule. Instead, they continue around the ring and are removed by their source node. Obviously, no spatial reuse for multicast or broadcast traffic is possible with this strategy. Source stripping is also used as a safety mechanism. If for some reasons unicast frames were not stripped by the destination, they are removed by the source when they return. If the source node disappears from the ring while a multicast frame is in transit (or both the source and destination node for a unicast frame), no node would ever remove the frame. It would seemingly keep circulating on the ring forever. This is prevented by the *tll* field in the RPR header. The *tll* works as a maximum hop count, and is decremented every time the frame reaches a new node. If the *tll* reaches zero, the frame is discarded.

In summary, there are four different conditions that lead to the stripping of a frame from the ring.

- If a bit error is detected in the received frame, the frame is discarded. For data frames, only bit errors in the header results in stripping.
- A unicast frame reaches its destination node.
- A frame returns to its source node, after traveling around the ring.

- The *tll* value of the frame reaches zero.

Frames that do not meet any of these conditions are passed on to the next downstream node.

2.2.2 RPR MAC Control Sub-layer

While the MAC data path sub-layer is responsible for the actual sending of frames from one node, the MAC control sub-layer is concerned with ring-wide functionality that monitors the traffic on the ring and takes appropriate action when an exception occurs.

Figure 2-4: Important functions of MAC Control sub-layer

Figure 2-4 shows an overview of the MAC control sub-layer and the functions it performs. The following sub-sections describe each of these functions.

2.2.2.1 Ringlet Selection

Before a frame can be transmitted on the ring, it must be decided which ringlet to transmit on. When this decision has been made, the frame can be put in the appropriate add queue. RPR allows this decision to be taken by the MAC client. If the MAC client has no preferences, a default ringlet will be chosen by the MAC control sub-layer. The algorithm used to choose the default ringlet is implementation specific. The only constraint imposed by the standard is that the same ringlet must be chosen for all frames with the same [*source address, destination address, traffic class*] tuple.

2.2.2.2 Topology Discovery

RPR contains a topology discovery mechanism in the MAC control sub-layer. This mechanism is responsible for providing each node with an overview of the number and placement of nodes on the ring. Every node keeps and maintains such an overview, called the topology image. The topology image consists of one record for each node on the ring. These records contain a hop count to that node on each of the ringlets. The records also contain additional information concerning each node, like protection status and reserved bandwidth status. The topology information is distributed to the nodes by broadcasting protection messages on the ring. A master node does not govern this distribution; Each node is responsible for initiating the protocol as needed.

Initially, a node's topology image contains only one record, with information about the node itself. It then triggers the topology algorithm by broadcasting a protection message, and continually listens for received protection messages. Each node that receives this protection message, responds by broadcasting a protection message on all

ringlets. By looking at the received protection messages, each node gets information about the distance to every other node, and information about its protection status. This information is used to build the topology image. The calculation of the topology image is repeated in case of a protection event, that is when a node discovers a failure in the connection to one of its neighbors. The topology calculation is also repeated periodically. The time between each periodic topology image update is configurable, the default period being 10 *msec*.

2.2.2.3 OAM

The Operation, Administration, Maintenance (OAM) functionality in the MAC control sub-layer offers a set of control functions to support configuration management, fault management and performance management. This includes functions for monitoring the connections between nodes, and for reporting abnormal behavior to a managing system. This information is distributed using special OAM control frames. The RPR standard allows some of these frames to be vendor specific, meaning that each individual vendor can define their own OAM frames and functionality.

2.2.2.4 Protection

RPR can transport diverse traffic types including delay sensitive traffic such as real time interactive audio. As a result RPR should be able to detect and repair a node crash or a link failure within the audio time constraint of 50 *msec*. A connectivity failure, either in a node or on a link, is termed a *protection event*. RPR offers two separate protection mechanisms to restore connectivity: *steering* and *wrapping*. Steering is the normal

protection mechanism, and is supported by all nodes. Wrapping is the optional mechanism in the RPR standard.

Each sender node is responsible for avoiding points of error. The node that first detects a protection event will send an error report around the ring. This is used by the other nodes to update their topology image, using the topology discovery mechanism described later. A new host-to-ringlet mapping is built in all the nodes. When this new image is complete, the point of failure can be avoided by choosing the correct ringlet to send on.

Figure 2-5: Normal operation versus Wrapped state.

If steering is the only protection mechanism in use on the ring, all frames that are sent from the time failure occurs to the time topology image is rebuilt, are lost. The advantage of the other protection mechanism in RPR, wrapping, is that it reduces the number of lost frames during a protection event. In case of a node failure, frames

destined for a node beyond the point of error are looped and placed on the opposite ringlet. This way, frames that are in transit when an error is detected can still reach their destination on the other ringlet. An example of wrapping is shown in Figure 2-5. We see that during normal operation, node 7 sends to node 5 with path 7-6-5, while in the wrapped state, the frames are sent via 7-8-1-2-3-4-5.

Wrapping increases the complexity of the ring as it introduces frame duplication and reordering. In Figure 2-5, consider a case when the ring returns from wrapped to normal state, or when steering causes frames from node 7 to node 5 to be transmitted on the outer ringlet. This transition suddenly reduces the number of hops the frames must travel, leading to the possibility that frames reach their destination in a different order than they were sent. Such frame reordering is not acceptable to some applications. RPR therefore allows traffic to be sent in two different modes, known as *strict mode* and *relaxed mode*. Frames in strict mode are never wrapped. This guarantees that no frame reordering will occur, but increases the number of frames lost during a protection event. Strict mode frames will be discarded at the point of error, and all such frames will be lost until the steering mechanism restores the connectivity. If a link fails or a node suddenly crashes, loss of some frames is inevitable without buffering. Frames in transit in a node that fails, e.g., because of a power failure, will be lost. The use of wrapping minimizes the number of frames lost. Note that during normal operation, RPR guarantees that no frames are lost or reordered on the ring; During protection event, frames may be lost and reordered.

2.2.3 Starvation Problem and Fairness Control

The RPR MAC is based on Buffer Insertion Ring (BIR) [24] as shown in Figure 2-2 where the insertion buffer has non-preemptive priority over the node buffer. The transit buffer(s) of the ring in RPR corresponds to the *Insertion Buffer* in BIR. With BIR, a node is prevented from adding traffic to a ringlet when its transit buffer is filled above a configured threshold. If a node continuously receives frames from aggressive upstream neighbors, it suffers so-called **starvation problem**. This unfairness is a natural by-product of priority scheduling used in the RPR MAC. Because of their relative location in the ring, some nodes may succeed in capturing a larger share of resources than others, and thus enjoy preferential treatment.

Procedure of the RPR feedback control to ensure fair access:

Step 1: Detect starvation.

Step 2: Estimate fair rate.

Step 3: Advertise the fair rate to all upstream nodes "hop-by-hop".

Step 4: Upstream nodes apply the fair rate to their token buckets.

Figure 2-6: Illustration of RPR Fairness Algorithm

Figure 2-6 shows a part of a ring with three nodes connected to each other. It illustrates the situation where node $(i+2)$ is prevented from transmitting on a ringlet, because node i is sending a continuous flow of frames through node $(i+2)$. If node $(i+2)$ were allowed adding traffic in this situation, the result would be buffer overflow and loss of frames in the transit buffer. This would not be acceptable, since RPR is designed to be a lossless network technology, meaning that no frames are lost on the ring during normal operation. As a result, a node may only add a frame if the transit buffer is empty. Thus, it is very easy for a downstream node to be starved by upstream ones. In RPR, the solution to the starvation problem is to force all stations to behave according to a specified fairness algorithm.

The objective of the fairness algorithm is to distribute unreserved bandwidth fairly among the contending stations so that they can use this bandwidth to send FE traffic. When the bandwidth on the transmit link of a node is exhausted, the link and node are said to be starved, or congested, and the fairness algorithm starts working.

RPR fairness assumes two modes of operations: *Aggressive mode* (AM) and *Conservative mode* (CM). Figure 2-6 shows that the RPR fairness algorithm (AM) works by sending a fairness message to the upstream nodes. The fairness control process requires 4 steps: starvation detection, fair rate estimation, hop-by-hop rate advertisement and fair rate application. Every node checks its congestion status periodically based on the conditions specific to its operation mode. Once congestion is detected, the congested node *estimates a fair rate* and sends a feedback control message including the *fair rate* to the upstream nodes in the opposite direction. Each upstream node applies the *fair rate* as the token generation rate of its token bucket to limit its *add rate* to the *fair rate*.

In AM mode, the initial *fair rate* is set to the *add rate* of the node traffic from the starved node. The underlying logic is that all nodes should have the same *add rate*. If the throughput of a starved node drops, all other nodes should follow. Once the starvation is clear, the *fair rate* ramps up to achieve better bandwidth utilization.

In CM mode, the *fair rate* is set to be the unreserved bandwidth divided by the number of *active nodes*. A node is considered to be active if it has at least one packet passing the starved node during the last interval. After the starvation clears, the *fair rate* gradually ramps up.

2.2.4 RPR Node Architecture

Figure 2-7: RPR node architecture from fairness algorithm viewpoint.

Figure 2-7 shows a simplified RPR node architecture with the main functional blocks related to the fairness algorithm. Only one direction of the ring is shown here, but the

reverse direction is similar. The node includes the following main modules that are related to the fairness algorithm.

- **Rate controller:** All ingress traffic entering the RPR ring passes the rate controller module. This rate controller is used for flow control. When a node receives an advertised fair rate from a downstream node, it has to enforce this rate through its rate controller. The rate controller consists of a token bucket. The credits in the token bucket are incremented based on the token generation rate at every time interval. A packet at the head of the queue can access the ring if the number of credits in the token bucket is at least equal to the packet size. In that case, the number of credits is decremented by the size of the packet transmitted.
- **Traffic monitor:** This module measures the rate of both node traffic and transit traffic.
- **Fair rate estimation/feedback module:** Upon congestion, this module uses the measurements from the traffic monitor to compute a fair rate and advertises this rate to all the upstream nodes through a control message. Once an upstream node receives this control message, it adjusts its rate to the received rate.
- **Ingress Buffer (IB):** This module holds the node/ingress traffic. RPR implements Virtual Destination Queuing (VDQ) in IB to avoid Head-of-Line (HOL) blocking. This is analogous to what is performed at an input port of a switch to avoid HOL blocking [33].
- **TB:** As explained in Section 2.2.1.4, the RPR standard defines two implementations for TB: 1-TB and 2-TB. The 1-TB implementation uses one TB for both HP and LP

traffic while the 2-TB implementation uses separate buffers for HP and LP traffic.

The scheduler serves the IBs and the TBs based on strict priority.

- **Priority scheduler:** The other important module in the node is the priority scheduler that serves traffic from TB and IB. Non-preemptive priority is given to the traffic from TB.
- **Buffer management module:** In 2-TB implementation, there is a relatively large LP transit buffer. This module prevents the transit buffer overflow using feedback control upon threshold crossing. Details on the buffer management will be given in Section 4.8.

2.2.5 Performance Limitations of RPR MAC

In reference to the MAC design goals listed in 1.2.1, RPR fairness control algorithm suffers from the following performance limitations:

1) Bandwidth Oscillations

The results in [27] [34] have shown that RPR fairness algorithm is prone to severe and permanent bandwidth oscillations when bandwidth requirements of the ring nodes are unbalanced⁵. The reason is that a congested node has no information of how much bandwidth each upstream node requires. Thus, the estimated fair rate does not accurately reflect the bandwidth requirements of the ring nodes and hence, nodes oscillate in search of the optimal fair rates. We demonstrate this oscillation phenomenon with the following illustrative example.

⁵ Unbalanced is referred to a case when all the sources contending over a segment of a ring have different bandwidth requirements. This diversity stems from the non-uniformity of the add/drop traffic to/from the ring.

Figure 2-8: Unbalanced traffic demands

Example 2-1: Let us use the classical “parking lot scenario” that has been described in [27] and we depict it in Figure 2-8. There are two flows competing over a congested link 2-3 with total bandwidth of 622 Mbps. Flows 1-3 and 2-3 have constant demands of 622 Mbps and 5 Mbps, respectively. This unbalanced demand creates a bottleneck at the link 2-3. Since the aggregated traffic rate at link 2-3 exceeds the link capacity, congestion is detected by node 2.

With RPR-AM, node 2 sends a feedback message to node 1 indicating the add rate of node 2 is 5 Mbps. Node 1 then limits its rate from 622 Mbps to 5 Mbps. At this point, the rate on the link 2-3 becomes 10 Mbps and hence, the congestion clears. Subsequently, node 1 increases its rate, up to 617 Mbps, based on a ramp-up scheme [15], e.g., low-pass filtering, until congestion at node 2 happens again. Repeating this cycle, the traffic of node 1 permanently oscillates between the maximum rate of 617 Mbps and the low rate of 5 Mbps as shown in Figure 2-9.

Figure 2-9: Simulation results demonstrating bandwidth oscillations with RPR-AM

Unbalanced traffic is also problematic for RPR-CM. With this mode, the advertised rate is calculated based on the number of active nodes at the time of congestion. For the given example, CM mode estimates the fair rate to be 311 Mbps since there are two active nodes contributing to the congestion. At this point, the total traffic rate at node 2 is 316 Mbps, so that congestion clears. Node 1 then increases its sending rate up to 617 Mbps when congestion happens again at the link 2-3. Repeating this cycle, the rate of node 1 oscillates between 311 Mbps and 617 Mbps.

2) Throughput loss

Throughput loss in RPR is a consequence of two phenomena: bandwidth oscillations and lack of bandwidth sharing [27] [28].

The impact of bandwidth oscillation on throughput loss can be seen from Example 2-1 described in the previous subsection. In this example, RPR-AM first estimates the fair

rate as the *add rate* of node 2, i.e., 5 Mbps. Although the rate of 5 Mbps is fair to both nodes, the utilization at link 2-3 is very low. The total bandwidth used for the two flows is only 10 Mbps, and hence, 612 Mbps of bandwidth is wasted. Our simulation results showed that the throughput loss can be up to 30% for the 2-flow case in Example 2-1. The result is described in Chapter 4.

If bandwidth is not shared properly, throughput loss occurs. *Bandwidth sharing* implies that the unused portion of the bandwidth allocated to one node can be reclaimed by other nodes. The allocated bandwidth may be unused either due to the lack of demand or because the flow is bottlenecked elsewhere. However, RPR does not have an efficient bandwidth sharing mechanism to satisfy the dynamic demands of different nodes. As in Example 2-1, the initial fair rate is estimated as 311 Mbps with RPR-CM mode while node 2 only requires 5 Mbps and cannot use all the allocated bandwidth of 311 Mbps. The unused bandwidth of 306 Mbps is wasted because it cannot be reclaimed by node 1 that has high demand of 622 Mbps. Also, throughput loss can occur if one node reduces its demand below fair rate and other nodes cannot utilize the unused bandwidth.

3) Slow Convergence

Convergence time represents the time it takes for a node to reach a steady state fair rate. RPR fairness algorithm starts with an estimated initial fair rate and, over many fairness control cycles, it attempts to reach the optimal fair rate that maximizes bandwidth utilization. Note that converging to the optimal fair rate may never occur when there is significant variability in the traffic demand of any node that contributes to congestion.

The feedback control nature of the RPR fairness algorithm implies that there is delay between the time the fair rate is calculated and the time it is taken into effect. The results in [35][36] showed that the convergence time is a function of link propagation delay and total delay.

A potential problem of large convergence time is that the traffic pattern can change before the fair rate converges. The system may never reach steady state if this happens. A sudden decrease in bandwidth usage of a connection can potentially increase the usage of other connections that share bandwidth with that connection. However, sources can increase their bandwidth usage to the new fair allocation only after a propagation delay. Thus, in a network where usage patterns change rapidly with time, even if the fair rate is estimated accurately at a downstream node, source may not receive the fair allocation on time, i.e., by the time a source adapts to the current allocation, the fair rate may have already changed. We present quantified performance results on convergence time in Chapter 4.

4) Large Access Delay

Access delay is defined as the delay from the time a packet arrived to a node till the time it gets on the ring. A good fairness algorithm should allow every node access its fair share of ring bandwidth with minimal delay.

In RPR, however, a node could be prohibited from accessing the ring, i.e., experience starvation, for a long time if upstream nodes send excessive traffic. The starved node can only access the ring bandwidth when the fairness control scheme takes into effect. Because of the closed loop feedback nature of the RPR fairness algorithm, there is a delay between the time the fair rate is calculated and the time it comes into

effect [37][38]. This delay, at minimum, is the summation of the round trip time (to allow the fairness messages reach all the upstream nodes) and the processing time at each node. We explore the access delay performance of RPR through both analysis and simulations in Chapter 4.

2.3 Enhancements to RPR Fairness Algorithm

Over the past few years, several proposals were made to address the limitations of RPR [27][35][39][40][41]. We describe them in this section.

1) Distributed Virtual Time Scheduling in Rings (DVSR)

DVSR [27] aims at addressing the first three limitations of RPR: bandwidth oscillations, throughput loss and large convergence time. The idea behind DVSR is that each node conducts *remote fair queuing*. This means that each node remotely controls its upstream nodes by informing them to send packets at the rate they would receive in a General Process Sharing (GPS) system [42], shown in Figure 2-10.

Figure 2-10: Illustration of remote fair queuing: (a) GPS server (b) DVSR approximation

To achieve this approximated fair rate control, DVSR needs to know the traffic demand of each node. However, this information is difficult to obtain due to the *distributed* nature of the bandwidth allocation problem, traffic dynamics and link propagation delay. Instead, each node uses per ingress byte counters to measure the arrival rate from the upstream nodes and its own add traffic as illustrated in Figure 2-11.

Figure 2-11: Measurements of the arrival rate (A_i)

DVSR implicitly assumes that the measured arrival rates represent the demands from the upstream nodes. The fair rate is then calculated based on the estimated demands from the measurements obtained during each control interval T .

Let us consider an arbitrary node n . Assume that there are k nodes sending traffic to node n , including the local traffic of node n . Assume the rates from the k nodes are ordered in a downward decreasing order, i.e., $A_1 \leq A_2 \leq \dots \leq A_k$, where the subscripts of the rates indicate the order, not the indexes of the transmitting nodes. The DVSR fair rate

at node n , F^n , is computed using the pseudo code listed in Figure 2-12. The algorithm is based on the following recursive equation:

$$F^n = \begin{cases} \max(A_i) + (C - \sum_i A_i) & \text{if } \sum_i A_i < C \\ (C - \sum_{A_i < F^n} A_i) / I & \text{otherwise} \end{cases}, \quad (2.1)$$

where C is the available (unreserved) link bandwidth and I is the number of flows whose rates are greater or equal to F^n , i.e., $I = |\{i : A_i \geq F^n\}|$.

```

if ( $\sum_i A_i < C$ )  $F^n = A_k + (C - \sum_i A_i)$ 
else{
     $j = 1$ ;
     $F^n = 1/k$ ;
     $count = k$ ;
     $L = C$ ;
    while( $(A_j < F^n)$  and  $(A_k \geq F^n)$ ){
         $count--$ ;
         $L = L - A_j$ ;
         $F^n = L / count$ ;
         $j = j + 1$ ;
    }
}

```

Figure 2-12: DVSR fair rate calculation

When there is no congestion, the DVSR fair rate is simply the largest arrival rate A_k plus the unused bandwidth as shown in (2.1). When congestion happens, the algorithm computes the fair rate based on the *max-min* fairness scheme [43]. Note that there is a difference between the max-min fairness scheme and DVSR. In DVSR, the *max-min*

rates are not allocated to the upstream nodes *individually*, instead, *one* rate is advertised to all the upstream nodes. This rate is the *largest* rate obtained from the max-min calculation. When an upstream node receives this maximum rate, it makes one of the following two decisions based on its demand. If the demand is less than this rate, the demand is granted; otherwise, it is capped by this rate. By doing so, DVSR *approximates* the *max-min* fair rate allocation through this maximum fair rate advertisement.

DVSR mimics GPS scheduling and max-min fairness as it uses traffic measurement to calculate the fair rate at every control interval. As a result, DVSR offers some improvements in bandwidth oscillations, convergence time, and throughput compared to RPR [27]. However, it suffers from the following disadvantages:

- **Unfairness**

The unfairness in DVSR is a direct consequence of using the following two mechanisms. The first one is the mechanism of allocating unused bandwidth. When the bandwidth is not fully utilized, DVSR calculates the fair rate to be the maximum flow rate plus the unallocated bandwidth. This could lead to unfairness for the downstream nodes. Assume the link is not fully utilized and one node decreases its demand by δ , the fair rate then increases by δ . If the number of upstream nodes is N and every node increases their rates by δ , the resulting traffic increase at the downstream node is $N \cdot \delta$. This means that the fair share of the downstream node can be reduced by $N \cdot \delta$. The second one is its fair rate advertisement mechanism. DVSR sends the maximum fair rate obtained from max-min calculation to all upstream nodes in order to achieve maximum utilization. If every node has constant demand, there is no problem. However, if an upstream node i increases its demand, f_i , to the

fair rate F (assume $F - f_i = \delta > 0$), the downstream node may suffer a rate decrease of δ which is unfair.

- **Inaccuracy in demand estimation**

A downstream node that conducts fair rate calculation does not have the knowledge of how much bandwidth each upstream node really needs. Hence, it assumes the measured arrival rates can represent the bandwidth demands of the upstream nodes. This assumption is not always true because of three reasons: (1) The demands may change during the fair rate estimation time; (2) The observed arrival rate is subject to the constraints imposed by the advertised fair rate; (3) There are multiple transit buffers along the path. The measured arrival rate at a node is actually the output rate of the upstream transit buffer; not the sending rate at the source.

As result of the inaccurate demand estimation, the fair rate calculated may not reflect the true demands.

- **Large Complexity:**

DVSR is complex in three aspects:

- Computational complexity:* Fair rate calculation in DVSR requires sorting the set of arrival rates, A_i 's, in every control interval, T . The computational complexity for this operation is $O(N \log_2 N)$ where N stands for the number of nodes on the ring.
- Hardware complexity:* The requirement of a counter for every upstream node adds additional hardware complexity to the design of a node. The maximum number of counters at each node is 256.

- c. *System complexity*: Whether a node is congested or not, it needs to calculate the fair rate and advertise this rate in every control interval. This results in large processing overhead in DVSR.

2) Virtual Queuing (VQ)

In an attempt to improve the unfairness problem of DVSR, VQ [39] was proposed. A VQ node uses the same counters as in DVSR to measure the arrival rates of the upstream nodes. Moreover, VQ identifies the number of nodes sending below and above the fair rate during the previous control interval. The nodes with arrival rate below the fair rate are termed “*input-limited*” nodes since their rates are limited by the demands (input); not by the fair rate. The nodes with arrival rate of fair rate are termed “*rate-limited*” nodes since their rates are limited by the fair rate. As it will be illustrated in Example 2.2, VQ subtracts the total rate of the “*input-limited*” nodes from the available bandwidth. The remaining bandwidth is then divided by the number of “*rate-limited*” nodes. The result is assumed as the fair rate. This procedure is the same as *max-min* algorithm except that it does not require sorting of the arrival rates [44]. Instead, it applies a linear comparison between the arrival rate and the current fair rate. Also, the new fair rate uses the previous fair rate calculation which results in computational complexity of $O(N)$ in comparison with $O(N \log_2 N)$ of DVSR.

VQ reduces the unfairness problem of DVSR by addressing the first issue of DVSR. When there is an unused bandwidth of δ and there are N rate limited upstream nodes, VQ increases the fair rate by δ/N , i.e., the unused bandwidth is shared among the rate-limited nodes. While DVSR increases the new fair rate by δ . The results in [39] have shown that the unfairness at a downstream node is less compared to DVSR.

However, VQ has not fully addressed the following issues:

- **Unfairness**

Although VQ improves the fairness of DVSR by allocating the unused bandwidth in a more conservative manner, it does not address the second issue with the DVSR which is the rate advertisement. In fact, both DVSR and VQ use the same principle in advertising the largest max-min fair rate. This results in potential unfairness for downstream nodes if an input-limited node increases its demand to the fair rate.

- **Throughput loss**

VQ approximates the max-min allocation by omitting the recursive loop in order to reduce the computational complexity. However, this is done at the expense of throughput loss. Let us elaborate this further. Assume $F(k)$ is the fair rate calculated during the control interval k . If the measured arrival rate of node i is the same as $F(k)$, VQ considers it a rate-limited node. VQ then calculates $F(k+1)$. Assume $F(k+1) > F(k)$. If the demand of node i , f_i , is larger than $F(k)$ but smaller than $F(k+1)$, $F(k) < f_i < F(k+1)$, there is bandwidth waste equal to $(F(k+1) - f_i) > 0$, which directly contributes to the throughput loss. In contrast, DVSR avoids this situation as it uses a recursive loop in its max-min rate calculation. The following example further illustrates this scenario.

Example 2.2: Consider four flows sharing a link with normalized link capacity of 1. The flows have constant demands of 0.1, 0.1, 0.3 and 0.5 (shown in Figure 2-13).

Figure 2-13. Illustration for Example 2.2

Assume that at the first control interval, the fair rate is set to be $\frac{1}{4}=0.25$ and every node has to throttle to this rate. At the second interval, since the link utilization is $0.1+0.1+0.25+0.25 = 0.7$, the fair rate should be increased to achieve higher utilization. Based on the measurement results in the first interval, there are two flows below the fair rate and two flows sending at the fair rate, i.e., there are two “input-limited” nodes and two “rate-limited” nodes. The new rate is then calculated as following:

$$(1-0.1-0.1)/2=0.4.$$

However, 0.4 is not the optimal fair rate since it results in bandwidth utilization of only 90%, i.e., $0.1+0.1+0.3+0.4 = 0.9$, resulting in 10% of throughput loss. The reason is that VQ assumes both rate-limited nodes have equal demand of the link capacity of 1 while in fact the demands of the two flows are 0.3 and 0.5. It takes another control interval to reach the optimal (max-min) fair rate of 0.5, raising the utilization to 100%. When there are more flows dynamically changing their demand,

it takes VQ many iterations to reach the “optimal” fair rate that maximizes the bandwidth utilization.

- **Lack of minimum fair share guarantee**

VQ cannot guarantee the minimum fair share of C/N where C represents the link rate and N is the number of nodes sharing the link. This is because VQ uses the rate calculated in the previous control interval to derive the new fair rate. On the other hand, DVSR uses the recursive loop shown in Figure 2-12 with starting initial rate of C/N . This can be illustrated by the following example:

Example 2.3: Consider four flows sharing the output link of node 4. The demands of the flows from node 1, 2, 3, and 4 are: 0.1, 0.2, 0.3, 0.4, respectively.

In the first control interval, assume the fair rate is:

$$F_1 = 0.25.$$

In the second control interval, node 1 and node 2 are input-limited, while the other two nodes are rate-limited. Hence,

$$F_2 = (1 - 0.1 - 0.2) / 2 = 0.35.$$

Assume node 1 and node 2 increase their demand to the fair rate of 0.35. The measured arrival rates at node 4 become: 0.35, 0.35, 0.3, and 0.4. Note that node 4 is the local node with demand (arrival rate) of 0.4 but it cannot add anything to the ring because the three upstream nodes have used the entire bandwidth.

In the third control interval, only node 3 is input-limited; the other three nodes are rate-limited, hence, the fair rate is calculated as following:

$$F_3 = (1 - 0.3) / 3 = 0.23$$

The rate of 0.23 is smaller than the minimum fair share of 0.25. This under-estimation of the rate results in 8% throughput loss.

- **Large Complexity**

Although VQ reduces the computational complexity of DVSR by eliminating sorting operation, it still needs to perform a linear comparison between the fair rate and the arrival rate of all upstream nodes. In addition, VQ has the same hardware and system complexity as DVSR does.

3) **Low Complexity Distributed Bandwidth Allocation (LCDBA)**

Alharbi and Ansari [40][41] proposed a new algorithm, Low Complexity Distributed Bandwidth Allocation (LCDBA), which is also based on DVSR scheme, with the difference that it calculates the fair rate by a simple equation:

$$F_k = C / \widetilde{N}_k, \quad (2.2)$$

where F_k is the fair rate at node k , C is the available bandwidth, $\widetilde{N}_k$ is the effective number of active nodes at node k . The key issue of this algorithm is to find $\widetilde{N}_k$, which is not necessarily an integer.

Similar to DVSR, LCDBA uses counters to measure the arrival rates of the upstream nodes. At the end of a control interval, these rates are compared to the current fair rate in order to obtain the activity level of all nodes. If a node sends more than the fair rate, it is considered as fully active. If an upstream node sends less than the fair rate, it is considered *partially* active. The effective number of active nodes $\widetilde{N}_k$ is then determined as follows:

$$\widetilde{N}_k = \sum_{i=1}^N \min(1, \frac{B_i}{F_k}), \quad (2.3)$$

where B_i is the counter representing the arrival rate from the upstream node i and N is total number of nodes in the ring. By applying (2.3), the unused bandwidth by the partially active nodes can be re-allocated to other active nodes with higher demand.

This approach improves the utilization to some extent with less computational complexity. However, there are some shortcomings associated with this approach.

- **Slow Convergence**

Compared to DVSR and VQ, LCDBQ requires longer time to achieve the optimal fair rate through its simple calculation. We illustrate this using the Example 2.2, given earlier. There are four flows sharing one link with constant demand of 0.1, 0.1, 0.3 and 0.5. Assume the initial fair rate at the first control interval is $1/4=0.25$. At the second interval, the measured number of effective active nodes is $(0.1/0.25+0.1/0.25 + 1 + 1) = 2.8$. Thus, the new fair rate is $1/2.8=0.36$. Carrying out the same calculation for the third interval results in a new fair rate of 0.47. It takes a few more control intervals to reach to the optimal (max-min) fair rate of 0.5. This results in throughput loss, until the optimal fair rate is achieved.

- **Large Complexity**

Compared to DVSR, LCDDBA doesn't require sorting, however, it needs a record of how much each node is sending relative to the fair rate. It also needs to perform a *division* to obtain the degree of activeness of each node.

- **Unfairness**

Since LCDBA uses the same strategy as DVSR in advertising the fair rate, the downstream nodes are subject to unfairness as well.

4) Enhanced Conservative Mode

The original RPR Conservative Mode (RPR-CM) has poor convergence properties and brings starvation to downstream nodes. To improve the performance of RPR-CM, the authors in [35] proposed “enhanced conservative mode algorithm”. This approach relies on more accurate estimation of the fair rate, earlier detection of the congestion; and fast re-evaluation of the fair rate.

The contributions of this paper include: (1) determining via extensive simulations the frequency and accuracy of fairness round trip time (FRTT) measurement needed to both achieve acceptable system performance and avoid starvation of downstream nodes; (2) introducing a new medium threshold to provide an early detection of congestion, thus reducing the likelihood of starvation of the congested node; and (3) aggressively reducing the rate when a node is severely congested.

However, the performance gain achieved through these modifications is at the price of more complexity including higher control overhead and more frequent control messages, etc.

2.4 Summary

This chapter has described the state-of-the-art MAC schemes for packet ring networks. The RPR standard is a new network technology developed by the IEEE. Among RPR's most important properties are spatial reuse, resilience, service flexibility and bandwidth efficiency. However, a key challenge for RPR is the design of a fair MAC

scheme. The RPR MAC is based on the BIR which assign non-preemptive priority to ring traffic over node traffic. As a result, downstream nodes may suffer starvation problem and fairness algorithm is required to ensure fair medium access. The RPR fairness algorithm utilizes rate-based feedback control to provide fair medium access control to FE traffic. However, it has certain performance limitations. Several proposals have been made to address these limitations. These attempts offer some improvements at the expense of certain performance tradeoffs and implementation complexity.

CHAPTER 3 **Description of Virtual Source Queuing (VSQ)**

3.1 *Introduction*

In chapter 2, we described the state-of-the-art MAC schemes for packet ring networks including the RPR standard and its enhancements. A key part of the RPR standard is its framework for the provision of distributed fairness control for FE traffic. Fairness control is the main technical challenge of the RPR standard and it has been an active research topic because of the traffic dynamics and the distributed nature of the control.

In this chapter, we propose a new MAC scheme, called VSQ, which complies with the MAC design goals. We also design an effective flow control scheme, called OHB, in order to provide lossless transit paths for VSQ. We then study the properties and parameters setting of VSQ.

3.2 *Research Problem*

From the discussion presented in Chapter 2, it becomes evident that there are four problems with the RPR MAC and its rate feedback control mechanism. First, a source cannot determine an increase in available bandwidth except by sending at a slightly increased rate and then back off if feedback messages are received from downstream nodes. Thus, a sudden large increase in the service rate can be adjusted only after several round trip times. This is undesirable, particularly in high-speed networks, where the bandwidth delay product is large. Second, it takes a few round trip times to adjust to a sharp decrease in service rate. In the meantime, the bottleneck queue builds up. Third,

when there is a decrease in service rate, the source sends at very nearly the service rate, so the built up queues never shrink, and the network becomes more prone to packet loss. Finally, the rate-increase strategy tends to push the network towards congestion, since the source always tries an increased sending rate, until the rate can no longer be supported.

The above problems point to the need for a better rate control algorithm. However, we observe that the enhancements to the RPR fairness control are based on the following two approaches:

- Provide more frequent feedback control as in “enhanced conservative mode”;
- Advertise a high fair rate to achieve high utilization as in DVSR, VQ and LCDBA.

To certain extent, these approaches improve RPR performance on oscillation, throughput and convergence time at the expense of more control overhead, complexity and unfairness. However, these proposals failed to address two fundamental issues of RPR fairness control:

- 1) The first issue is the large access delay, which is a direct consequence of feedback rate control scheme. Theoretical results reported in [38] prove that oscillations are inevitable for a fairly general form of feedback based controls due to the delay in the feedback.
- 2) The second issue is the lack of bandwidth sharing as all the fairness control schemes use *one* fair rate for *all* upstream nodes. The upstream nodes have dynamic demands, which make it almost impossible to use one fair rate for all nodes and, at the same time, achieve maximum utilization. If one upstream node

decreases its demand by Δ , the unused bandwidth Δ cannot be used by other nodes, resulting in a throughput loss of Δ .

Table 3-1: Design goals compliance comparison

Design goals	RPR	DVSR	VQ
Fairness	Good	Poor	Average
High throughput	Poor	Good	Average
Low access delay	Poor	Good	Average
Simplicity	Good	Poor	Average

In summary, none of the existing MAC schemes can achieve the design goals described in Chapter 1. Table 3.1 shows a coarse comparison of the three schemes regarding the compliance with the MAC design goals. As seen from the table, tradeoffs have been made among the metrics, i.e., none of the schemes can simultaneously provide guaranteed fairness, high throughput, short access delay and low complexity. For example, DVSR improves the utilization by advertising high fair rate as opposed to the low fair rate used in RPR, but it suffers more unfairness compared to RPR.

This finding motivates us to re-examine the RPR MAC shown in Figure 2-3. RPR MAC is responsible for scheduling the access to the ring between the local add traffic and the transit traffic. RPR MAC uses priority-based scheduling policy: transit traffic has non-preemptive priority over local add traffic. Priority scheduling offers simple access control and an express transit path for the high priority traffic with stringent delay requirement. However, the priority scheduling is not suitable for FE traffic since it is unable to provide fairness. In fact, we believe the *priority scheduling* in RPR MAC is the

root cause of the unfairness problem with RPR [45][46]. However, none of the existing fairness control schemes has tackled this fundamental cause. Instead, they provided an overlay solution to alleviate the problem to certain degree by regulating the ring access through fair rate estimation and feedback control. The rate control based approaches cannot realize all the design goals. They doom to suffer from feedback delay and inaccuracy in rate estimation, leading to unfairness and throughput loss. The inherent unfairness problem of RPR MAC makes the following two questions worth answering:

- 1) Is there a method to provide fair medium access control different from RPR MAC?
- 2) Can we avoid the complex overlay fairness control schemes, if the new MAC can fundamentally provide fairness?

This brings us to our research topic: *fair medium access control scheme for metro packet rings*, which is illustrated in Figure 3-1.

Figure 3-1: Illustration of the research problem

The objective of the proposed research is to design a MAC scheme that fairly and efficiently utilizes packet ring bandwidth. With respect to the stated research topic and the state-of-the-art approaches, the two involved research issues are:

- 1) What is a suitable MAC scheme for metro packet rings?
- 2) What is the performance of the proposed MAC in comparison with that of RPR?

In the following sections, we detail our proposed approach and the methodology to assess its performance.

3.3 VSQ MAC Scheme

An ideal MAC scheme for packet rings not only assures fairness for FE traffic but also maximizes the throughput on each segment of the ring. The main idea of the proposed approach is to provide a fair share of the bandwidth to the competing nodes on every link they pass through.

The RPR standard defines that when the demand for bandwidth on a segment of the ring is greater than the supply, the bandwidth shall be divided fairly among the nodes sending packets over that part of the ring. Every link of a RPR ring works independently, so that simultaneous transmission is achieved. The addition of individual frames is not synchronized between ringlets and the frame transmission event on any link is independent of the frame transmission events on other links [15].

In summary, packet rings have a unique fairness requirement. The add traffic of each ingress node should receive a fair share of bandwidth on each link it demands [12] [27]. In other words, we can treat each link of a ring as a separate resource and divide the

bandwidth equally among nodes that share this link. Based on the above observations, we formulate the fair access control problem of packet rings and then propose our solution:

*Let us consider a uni-directional ring with N nodes numbered from 1 to N along the ring direction. For each node i ($1 < i < N$), we define L_i to be the bandwidth (or link capacity) of the outgoing link between node i and node $(i+1) \bmod N$. We call all the nodes that send traffic through L_i the **sources** of L_i . The problem of fair medium access equates to how to fairly allocate the bandwidth of L_i among all the competing sources.*

3.3.1 Description

To solve the above problem, we provide a virtual queue⁶ for traffic coming from each possible source L_i into the secondary (low priority) transit buffer. The maximum number of virtual queues at the transit buffer of each node is $(N-1)$ since a node does not send traffic to itself. Let $Q_i(j)$ ($1 \leq j \leq N$) represent the j^{th} virtual queue at node i . A weight $w_i(j)$ is assigned to $Q_i(j)$ based on the relative importance of that node. The option of setting different weights⁷ to different nodes could result in prioritizing the nodes and achieving differentiated QoS. We call this access control scheme “Virtual Source Queuing (VSQ)” [47]. The term “virtual” is due to the fact that all the queues share one physical buffer and the term “source queuing” is because the virtual queues are identified by their traffic sources. Any work-conserving scheduling algorithm that provides fair sharing, such as Weighted Fair Queuing (WFQ) [42], Token Bank Fair Queuing (TBFQ) [48], Credit-Based Fair Queuing (CBFQ) [49], Deficit Round Robin (DRR) [50], Weighted Round

⁶ The virtual queues can be implemented using link lists that share one physical buffer.

⁷ For simplicity of explanation, we consider all nodes have equal weight of 1, i.e., $q_i(j) = 1$ for $i, j = 1, \dots, N$.

Robin (WRR) [51] [52] or Self-Clocked Fair Queuing (SCFQ) [53], can be used to serve these virtual source queues. The choice of scheduling algorithm is implementation-specific and does not affect the framework of VSQ. In this thesis, we choose WRR due to its simplicity [54]⁸.

Figure 3-2: Illustration of VSQ

Figure 3-2 illustrates the VSQ datapath with multiple virtual source-based queues in the LP transit buffer. We mark the scheduling priority index as: HP transit traffic: 1, HP Node traffic: 2, LP transit and Node traffic: 3. Recall that in the RPR data path shown in

⁸ We assume the same average packet length for each source to simplify analysis. However, this assumption does not affect the validity of the framework of VSQ. If the average packet length is different, we can use DRR which eliminates the unfairness due to different packet lengths. Authors of [55] propose to use DRR to schedule the high priority node traffic and transit traffic to reduce the access delay of high priority node traffic. However, the congestion control (transit buffer overflow) issue with this approach has not yet been addressed.

Figure 2-3, the priority index is given as: HP transit traffic: 1, HP Node traffic: 2, LP has priority over LP Node traffic. We see that the HP traffic is treated the same way in both RPR and VSQ. The main difference between the two schemes is the way that LP traffic is served. In VSQ, the LP ring traffic and LP node traffic are placed in the queues associated with their source addresses. The scheduler serves all the source queues in a fair manner to allow every node receive a guaranteed fair share of the ring bandwidth. In RPR, however, LP ring traffic has priority over LP node traffic, which can lead to starvation and long access delay for LP node traffic. We can view the scheduler in VSQ node as a 2-tier scheduler: LP traffic is served with a fair scheduler and HP traffic (with reserved bandwidth) has priority over LP traffic.

Figure 3-3: Illustration of the traffic flow isolation with VSQ

Figure 3-3 shows the concept of VSQ in a unidirectional ring with four nodes. Every node has 4 virtual queues. For illustration purpose, we show only the LP traffic of node 1 sent evenly to all other nodes. In each node, LP traffic of node 1 is placed at virtual queue 1. As seen, traffic of each source has been isolated from each other. Thus, an ill-behaving source that sends excessive traffic may suffer delay or loss while a well-behaving source node has guaranteed access to their fair share of bandwidth.

A pseudo-code implementation of the generic VSQ MAC is shown in Figure 3-4, consisting of *Initialize*, *Enqueue* and *Dequeue* routines. The *initialize* function is invoked before the VSQ can be utilized, and it allocates memory for data structures and VSQ parameters. The *Enqueue* function is invoked when a packet arrives. If a packet belongs to an unknown flow, provision for a new virtual queue is necessary. This includes memory allocation to accommodate the parameters associated with the new flow, adjusting the total number of flows, etc. The fair scheduler serves the virtual source queues by using the *Dequeue* procedure.

3.3.2 VSQ Node Architecture

Figure 3-5 illustrates the architecture of a VSQ node. Compared to the RPR node architecture in Figure 2-7, VSQ eliminates three modules that are related to fairness control: fair bandwidth allocation module, traffic monitor and rate controller/Virtual Destination Queuing. The reason is that VSQ uses a fair scheduler to serve all the queues so that every node can have guaranteed fair share of the ring bandwidth without fair rate estimation and fairness feedback control.

Initialization:

Constants: $QUEUE_MAX$: maximum number of nodes on the ring;

Assign $w[i]$, weight of node i , $1 \leq i \leq QUEUE_MAX$

Enqueue: -Upon arrival of packet p :

$i = \text{source address } (p)$,

if ($\text{existsInActiveList}(i) == \text{FALSE}$) **then**

$\text{addToActiveList}(i)$; //add i to the tail of active list

$\text{allocate}(\text{queue}[i])$; // Allocate a queue

End if

$\text{enqueue}(i, p)$; //enqueue packet p to $\text{queue}[i]$

$qlen[i]++$; //increment the number of packets of $\text{queue}[i]$.

Dequeue:

While ($TRUE$) **do**

if $ActiveList$ is not empty **then**

$i = \text{HeadofActiveList}$,

$\text{RemoveHeadofActiveList}$;

$k = \min(w[i], qlen[i])$;

$qlen[i] = qlen[i] - k$;

serve ' k ' packets from $\text{queue}[i]$;

if $qlen[i] == 0$ **then**

$\text{removeActiveList}(i)$

else $\text{addToActiveList}(i)$;

end While

Figure 3-4: Pseudo code for VSQ

Figure 3-5: VSQ node architecture

We describe the remaining modules in the following.

- **IB:** Physically, it is the same as in the RPR node. However, RPR implements VDQ in the IB in order to avoid HOL blocking, while VSQ does not need VDQ as there is no HOL blocking.
- **TB:** As in RPR node, this buffer holds traffic arriving from all the upstream nodes. The difference in VSQ is that this buffer implements multiple virtual source-based queues. The LP ring traffic is placed into these virtual queues based on their source addresses. The added complexity of virtual source-based queues is offset by the elimination of VDQ at the IB of RPR. We also show that the total buffer size remains the same or less than RPR.

- **Fair Scheduler:** Unlike RPR which uses priority scheduler, VSQ uses fair scheduler to serve all the source queues in both the transit buffer and the node buffer. Since we assume WRR scheduler in this study, all the queues are served in a round robin fashion.
- **Transit buffer management module:** Similar to RPR, a buffer management module is required to avoid buffer overflow given the limited size of the transit buffer. We propose an efficient buffer management scheme called OHB scheme to ensure lossless transit paths. More details on this scheme are provided in the next subsection.

3.4 Flow Control Scheme for VSQ: OHB

RPR standard calls for lossless transit paths, i.e., once a packet gets on the ring, it cannot be lost during normal operation. In VSQ, the transit buffer consists of multiple source-based virtual queues and the scheme has the knowledge on the amount each source contributes to the occupancy of the transit buffer. This information can be utilized to avoid buffer overflow so that there is no need to enforce fair rates at the ingress points as RPR does. Based on this strategy, we propose a simple flow control scheme called OHB to avoid buffer overflow in VSQ [47].

3.4.1 Description

We use two sets of thresholds for each virtual source queue⁹: *VSQHighThreshold* and *VSQLowThreshold* ($VSQHighThreshold > VSQLowThreshold$). A threshold is indicated

⁹ For simplicity, we assume equal threshold and buffer allocation for all sources.

as been exceeded whenever the number of bytes queued in the virtual queue is greater than the threshold value. If every node sends at its fair rate, there is no queue build up and the feedback control scheme will not be activated. On the other hand, if a node sends traffic more than its fair share, its queue length grows and the flow control scheme can directly control this source queue.

Procedure of the VSQ feedback control to avoid buffer overflow:
Step 1: Detect potential buffer overflow of a certain source queue.
Step 2: Notify the neighboring upstream node to reduce the sending rate from that specific source queue.

Figure 3-6: Illustration of OHB scheme

Figure 3-6 illustrates the operation of OHB scheme. Let us consider the pair of adjacent nodes on the ring, node $(i+1)$ and node $(i+2)$. When queue i at node $(i+2)$, $Q_i(i+2)$, exceeds the upper threshold, $VSQHighThreshold$, a feedback control message $OFF[i]$ is sent from node $(i+2)$ to the upstream neighbor node $(i+1)$. $OFF[i]$ indicates that the traffic from queue i at node $(i+1)$, $Q_i(i+1)$, is excessive and it may cause buffer overflow at the downstream node $(i+2)$. This implies that flow i exceeded its fair share on link between node $(i+1)$ to node $(i+2)$, $L_{(i+1)}$. Therefore, node $(i+1)$ decreases the

transmission rate of flow i by *pausing* servicing packets from queue $Q_{i(i+1)}$ until it receives an $ON[i]$ signal from node $(i+2)$. This $ON[i]$ signal is triggered when the queue $Q_{i(i+2)}$ descends below the lower threshold, $VSQLowThreshold$. While the scheduler pauses the service rate of queue $Q_{i(i+1)}$, the utilization of link $L_{(i+1)}$ does not necessarily decrease. This is because the scheduler is work-conserving and the link is utilized if other queues are not empty.

```

-Upon arrival of packet  $p$  at node  $(n)$ :
   $i = \text{source address}(p)$ ;
  enqueue packet  $p$  to  $\text{queue}[i]$ ;
  if ( $q_{len}[i] > VSQHighThreshold$ ) then
    send  $OFF[i]$  to upstream neighbor node  $(n-1)$ 

-Upon departure of packet  $p$  from queue  $[i]$  at node  $(n)$ :
  if ( $q_{len}[i] < VSQLowThreshold$ ) then
    send  $ON[i]$  to upstream neighbor node  $(n-1)$ 

-Upon receiving  $OFF[i]$  from downstream neighbor node  $(n)$ 
  block( $i$ ); //block serving queue  $[i]$  at node  $(n-1)$ ;

-Upon receiving  $ON[i]$  from downstream node  $(n)$ 
  unblock( $i$ ); //resume serving queue  $[i]$  at node  $(n-1)$ ;

```

Figure 3-7: Pseudo code of OHB scheme

Figure 3-7 shows the pseudo code for the scheme. The OHB scheme is very simple since it only checks the queue length upon reception and transmission of packets. The checking of the queue length in OHB accounts for the same overhead as in RPR-2TB implementation. The backpressure control signal is efficient because not only it works fast but also it does not contain fair rate information. It only requires 9 bits, 8 bits is used to represent maximum 256 queues and one bit to identify the ON or OFF state. The control signal is sent on the opposite direction and can be piggybacked onto a data packet

if any is available, or use some reserved fields of the data packets as the approached used in [56]. Thus, the overhead for carrying these control signals is minimal. The backpressure control signal can be propagated, when it is necessary, on a hop-by-hop basis to the source node that sends more than its fair share¹⁰. It is then up to the source node to control its sending rate by queuing the traffic in the node buffer.

On the other hand, RPR feedback control is more complex because it is used not only to prevent packet loss but also to enforce fair access. There are four steps involved in the RPR feedback control operations as described in Section 2.2.1 (Figure 2-6). Because RPR has no knowledge of which source contributes to the buffer overflow, it has to notify *all* the upstream nodes to slow down through fair rate estimation and feedback mechanism. The fairness messages used in RPR has to include the fair rate information and the source address of the node that initiates the fairness message. RPR fairness frame has 16 bytes and each node has to update its “fair rate” every control interval¹¹. The frequency of advertising fairness message is controlled by *advertisementInterval* that is defined by the RPR standard as [15, page 240]:

$$advertisementInterval = (8 * FAIRNESS_SIZE) / (lineRate * advertisementRatio), (3.1)$$

where the *advertisementRatio* is defined as the ratio of the link capacity reserved for fairness frames to the *lineRate*. The *advertisementRatio* has a range of [0.00025-0.01], which means that the bandwidth used for fairness control messages could be up to 1%. In

¹⁰ OHB is a framework for feedback control. We believe this approach is more effective than *n-hop* (where $n = 2, 3, \dots, N$) backpressure. This is because, intuitively, if the traffic burst length is long, then one-hop backpressure is applied to a number of hops through one-hop backpressure propagations. This allows throughput maximization. By contrast deploying, *n-hop* backpressure may lead to throughput loss as the burst length may be smaller than the round trip time through *n-hops*.

¹¹ RPR standard defines a default value for the control interval: 100 μ s for 622 Mbps link.

addition, there is also overhead associated with the processing of the fairness control messages.

The simplicity in VSQ feedback control also results in less transit buffer requirement compared to RPR as we show later in Chapter 4. This is because at any point of time, the OHB scheme tries to evenly *distribute* the packets of a flow across the corresponding buffers on a path from its source to its destination. While RPR lets the buffer *concentrates* at the congested node, and hence, it requires every node to have a larger buffer than in VSQ to prevent buffer overflow. Table 3-2 compares the feedback control for RPR and VSQ.

Table 3-2 Comparison of feedback control for RPR and VSQ

	RPR	VSQ
Objective	<ul style="list-style-type: none"> Prevent transit buffer overflow Enforce fair rate 	<ul style="list-style-type: none"> Prevent transit buffer overflow
Control message content	16 bytes include source address and fair rate information	9 bits, include source queue number and ON/OFF indication.
Frequency for control messages	<ul style="list-style-type: none"> Every control interval (e.g., $100\mu s$), every node updates and sends its fair rate; Every advertisement interval ($100\mu s$), every node advertises the rate it receives. 	Only upon threshold crossing/congestion. When there is no congestion, there is no feedback.
Response time for control message to take into effect	Multi-hop delay: from the congested node to the source, receive the fair rate and enforce the fair rate	One-hop delay: the blocking the service of one source queue only takes one-hop-delay
Buffer size requirement¹²	<ul style="list-style-type: none"> Buffer is <i>concentrated</i> at the congested node Buffer should be large enough to hold all the transit traffic during the response time which is large. 	<ul style="list-style-type: none"> Buffer is <i>distributed</i> along the path. Buffer should be large enough to hold all the transit traffic during the response time which is small.

¹² Detail analysis on transit buffer size requirements is given in Section 4.8.

There are some important differences between OHB scheme and previous flow control schemes [57][58][59][60][61]. We compare OHB with two main categories: Hop-By-Hop (HBH) flow control [62] and ON-OFF flow control [63]. It is noted that the OHB scheme employs the principle of ON-OFF flow control. However, the difference lies in that the OFF signal is sent only to the “guilty” source queue with OHB as opposed to blocking the whole scheduler with ON-OFF flow control. Autonet, a high-speed local area network, uses an ON-OFF type of flow control mechanism between nodes [63]. However, it is a “*source-blind*” scheme meaning it does not have per source queuing. As a result, it causes unfair service to users, spreads congestion to other nodes and causes oscillations in the queue occupancy at the bottleneck. Because of these factors, the control mechanism used in Autonet does not appear to be useful for wide area or metropolitan area high-speed networks.

OHB implicitly controls the rate so that there is no need to calculate the rate explicitly. While the HBH rate control [64] would require an accurate rate calculation at the bottleneck for each flow at each cycle. Other window-based HBH schemes [43] would require accurate state information about each connection, such as the sequence number of packets, to be available at the switch. In addition, they do not distinguish between connections and requires the control of a window size. For quick reference, Table 3-3 compares the difference between OHB and other flow control schemes.

The advantage of OHB is a natural result of utilizing the source information in the VSQ scheme. We can use Figure 3-6 to illustrate this advantage. Assume node $(i+2)$ is congested and the queue level in source queue i exceeds the threshold. This triggers an $OFF[i]$ signal from node $(i+2)$ to node $(i+1)$. Once node $(i+1)$ receives this backpressure

signal, it only blocks the transmission of virtual source queue (i). This indirectly penalizes the traffic from source (i) to node ($i+2$).

Table 3-3: Comparison of flow control schemes

	Explicit or implicit rate control	End-to-end or hop- by-hop	Source blind or source aware
ON-OFF	Implicit	Hop-by-hop	Source blind
HBH	Explicit	Hop-by-hop	Source blind
VSQ-OHB	Implicit	Hop-by-hop	Source aware

3.4.2 Analysis

In this section, we conduct some analysis to show that OHB is an effective flow control scheme. Let us first prove the following propositions for ease of analysis.

Proposition 3.1: *Consider traffic arriving at an initially idle node i with a rate r , let s_i denote the service rate at node i .*

- (a) If $r \leq s_i$, there is no queuing at node i , and the departure rate from node i is r .
- (b) If $r > s_i$, there is queuing at node i , and the departure rate from node i is s_i .

Proof:

- (a) Initially, since the node is idle, its queue is empty. If the first packet arrives at time t_0 , it will depart at time $t_0 + 1/s_i$. Packets in the arriving stream are spaced $1/r$ time units apart. Thus, the next packet arrives at time $t_0 + 1/r$. Since $r \leq s_i$, $1/r \geq 1/s_i$ and $t_0 + 1/r \geq t_0 + 1/s_i$, the next packet arrives only after the first one has left. Thus, there is no queuing

at the server. The departure rate of the packets is constrained only by the arrival rate, and hence the output stream from the server has a rate r .

(b) Since the departure of the first packet happens after the arrival of the next packet, the second packet will be queued in the server. If there is a queue already, and a packet arrives before the departure of the previous packet, it will only add to the queue. Since the departure stream from the server has an inter-packet spacing of $1/s_i$, the output stream is at rate s_i .

Proposition 3.2: *Consider two adjacent nodes i and $i+1$. If data enters node i at a rate r such that $s_i \geq r > s_{i+1}$, queuing occurs only at the node $i+1$.*

Proof:

Since $r \leq s_i$, there is no queuing at node i (using Proposition 3.1-(a)). Hence, the departure rate of packets from node i , as well as the arrival rate at node $i+1$ is r . Since $r > s_{i+1}$, there is queuing at the node $i+1$ (using Proposition 3.1-(b)).

Note that in our model, it is not possible to have more than one bottleneck. While queuing may occur at more than one node, the service rate of a flow is determined by the lowest indexed node with a service rate of μ , and this will be the bottleneck.

The motivation of OHB flow control scheme is based on the following observations:

- 1) VSQ allows each node to have the knowledge of which sources exceeds its fair share.
- 2) Based on Proposition 3.1, if a source sends at the rate of the bottleneck fair rate, it will have no queue.

- 3) Based on Proposition 3.2, if a source sends at the rate higher than the bottleneck fair rate, the queue grows at the bottleneck.
- 4) If a node with large buffer notifies its upstream neighbor, the information will propagate to the source. *Independent* transmission over nodes allows flow control to be exercised at each node along the path of a traffic stream.
- 5) One of the special features of packet rings is that each node on the ring acts as *both* a source and a switch, i.e., it has both the function of adding and passing traffic. Hence, it is possible to apply one-hop backpressure flow control. While with a regular switched network, one-hop control is infeasible because sources and switches are separate.
- 6) With OHB, intermediate nodes play an active role in regulating traffic. This is feasible in packet ring networks with VSQ MAC where each node has the source information through virtual source queues. While RPR feedback control is “*source-blind*” regulation mechanism resulting in unnecessary blocking of well-behaving sources. Each intermediate node controls the sending rates for individual traffic streams. OHB is a decentralized scheme where only the feedback received from the nearest neighbor is used by the node for regulating network traffic. As a result, OHB has shorter delay in its feedback cycle compared to the end-to-end round trip time in a ring. The reduced lag time leads to more responsive and accurate control of traffic.

The primary goal of the OHB is to prevent buffer overflow and match the sending rate of a source to the service rate observed at the downstream bottleneck link. The rate adaptation procedure also attempts to maintain a queuing of waiting packets at each

intermediate node. This queuing allows a flow to quickly increase its sending rate when any additional capacity becomes available. The desired number of packets waiting at each hop, referred to as a buffer threshold, is a user-specific parameter. We examine in Section 3.6 how to tune this parameter to trade-off buffer size and control overhead.

OHB has certain advantages, particularly in networks with a large propagation delay-bandwidth product. Since VSQ provides the source information, OHB seems to be an optimal choice. Finally, because OHB restricts traffic overloads to the periphery of a network, well-behaving nodes are protected from ill-behaving nodes. The simulation results in Chapter 4 support these intuitive arguments.

3.5 *Properties of VSQ*

In this section we present a detailed analysis of the performance characteristics of the VSQ MAC scheme based on the following important properties.

- Rate guarantee: VSQ provides a minimum rate guarantee for each flow on the ring.
- Fairness bound: VSQ ensures the rate difference between two flows is bounded.
- Access delay bound: VSQ offers bounded access delay, i.e., bounded waiting time for a node to access its fair share of ring bandwidth.
- Isolation: VSQ provides a firewall protection among all flows.

Before discussing these properties, we introduce some formal definitions used in this section.

Definition 3.1: Define a *flow* as a sequence of packets generated by the same node. At any link of an N -node ring, there are maximum N flows sharing the link. Each node is assigned a weight w_k ($1 \leq k \leq N$). The flow originated from node k has a weight of w_k .

Definition 3.2: Define *active flow* as a flow that has at least one packet awaiting service or being served during a certain time interval. *Active List* is a list of active flows.

Definition 3.3: Define *backlogged flow* as a flow that has packets awaiting transmission or already being transmitted.

Definition 3.4: Define $T_{\text{session}}(k)$ as the duration of the time period that flow k has one or more backlogged periods.

Definition 3.5: Define $\text{Sent}_k[t_1, t_2]$ as the service offered, i.e., amount of traffic transmitted by flow k ($k=1, 2, \dots, N$) during the interval $[t_1, t_2]$, $0 < t_1 < t_2 < T_{\text{session}}(k)$.

Definition 3.6: Define a *round* as a round robin iteration during which the WRR scheduler serves all the flows that are present in the *Active List*.

Definition 3.7: Define $W(i)$ as the sum of the weights of all active flows that are being served by the WRR server at node i . W_{sum} is the sum of the weights of all nodes.

Definition 3.8: Define M as the size (in bits) of the largest packet. Define m as the size (in bits) of the largest packet that is actually served during the execution of a scheduling algorithm. Note $M \geq m$.

3.5.1 Fair Rate Guarantee

We show that VSQ offers a guaranteed rate service for each flow on the ring. Each node can access its guaranteed rate with minimum delay.

Proposition 3.3. *For a backlogged flow k ($k=1,2,\dots,N$) that is served under VSQ with a weight of w_k and the link rate of C , the guaranteed minimum service rate is $r_k = (w_k/W_{sum}) \cdot C$.*

Proof:

Since each source-based virtual queue is served independently, the backlogs in the queues are therefore served independently. A backlogged packet in a source queue cannot be affected by the backlogs from other source queues.

According to the Definition 3.7, a flow k has a weight of w_k and the total weight of all the active flows passing link i is $W(i)$. Thus, at node i , flow k receives service rate of

$$r_k(i) = (w_k / W(i)) \cdot C. \quad (3.2)$$

Because the maximum value of $W(i)$ is W_{sum} across all the links along the end-to-end path, flow k can receive the minimum service rate of

$$r_k = \min_{i \in (1,\dots,N)} r_k(i) = (w_k / W_{sum}) \cdot C. \quad (3.3)$$

3.5.2 Fairness Bound

To evaluate the fairness of VSQ scheme, we deploy the widely used throughput fairness measure called Fairness Bound (FB) introduced in [53]. FB measures the worst case difference between the normalized service received by different flows that are backlogged during any time interval. It is clear that there is no need to compare a flow that is not backlogged with the backlogged ones because the former does not receive any service. If the value of the FB is small, it means that the service discipline closely emulate a bit-by-bit round-robin fair server.

Definition 3.8: We define FB as

$$FB = \max_{i,j \in (1 \dots N), 0 < t_1 < t_2 < T_{\text{session}}} \left| \frac{Sent_i[t_1, t_2]}{w_i} - \frac{Sent_j[t_1, t_2]}{w_j} \right|,$$

where $Sent_i[t_1, t_2]$ is defined (in Definition 3.5) to represent the service received by flow i during $[t_1, t_2]$, $0 < t_1 < t_2 < T_{\text{session}}$ and $T_{\text{session}} = \min(T_{\text{session}}(i), T_{\text{session}}(j))$. FB represents the maximum absolute value of the bandwidth difference between all pairs of flows i and j , $i, j = 1, 2, \dots, N$. The smaller the FB is, the better the fairness algorithm would be [81]. For ideal fairness, FB is zero whereas for any packetized system, FB is bounded away from zero by a constant.

With WRR scheduling, each flow i has an weight w_i . The server visits all flows in a predetermined order. When the server visits a source queue i , it serves up to a maximum of w_i packets in a first-come-first-served manner.

Let n_i and n_j denote the total round-robin rounds received by two continuously backlogged flows i and j in the interval $[t_1, t_2]$, respectively. Without loss of generality, we assume that flow i starts receiving service before flow j does, thus, we have

$$n_i \leq n_j + 1. \quad (3.4)$$

Using definition 3.6, we have the following for flow i ,

$$Sent_i[t_1, t_2] = n_i w_i, \quad (3.5)$$

and similarly, we have for flow j ,

$$Sent_j[t_1, t_2] = n_j w_j. \quad (3.6)$$

Combine (3.4), (3.5) and (3.6), we obtain:

$$\left| \frac{Sent_i[t_1, t_2]}{w_i} - \frac{Sent_j[t_1, t_2]}{w_j} \right| = \left| \frac{n_i w_i}{w_i} - \frac{n_j w_j}{w_j} \right| = |n_i - n_j| \leq 1. \quad (3.7)$$

Based on (3.7), we have the FB for VSQ as:

$$FB_{VSQ} = \max_{i, j \in (1 \dots N), 0 < t_1 < t_2 < T_{session}} \left| \frac{Sent_i[t_1, t_2]}{w_i} - \frac{Sent_j[t_1, t_2]}{w_j} \right| = 1. \quad (3.8)$$

3.5.3 Access Delay Bound

Definition 3.9: Define *access delay bound* as the maximum time between the instant the first packet of a new flow arrives in its queue and the instant the packet is scheduled.

A small bound on the access delay guarantees that the scheduler serves a new flow or an active flow within an acceptable finite time. In the following theorem we derive the VSQ bound.

Theorem 3.1. *For a VSQ node that serves n active flows with link rate C , the access delay, D , of a newly active flow has an upper bound given by*

$$D \leq \frac{MW_{sum}}{C}.$$

Proof:

When the first packet of a newly active flow arrives at the node queue, the flow is placed at the tail of the Active List and is served after all the n previously active flows are served. When WRR server visits queue i , it serves up to w_i packets (Definition 3.1). We use $Sent_i(k)$ to denote the number of packets that are transmitted from flow i in the k^{th} round. W_{sum} is the total weight as defined in Definition 3.7. M is defined in Definition 3.8 as the maximum packet size. We have:

$$Sent_i(k) \leq Mw_i, \quad (1 \leq i \leq n) \quad (3.9)$$

Summing up the $Sent_i(k)$ over all the flows, we obtain:

$$\sum_i Sent_i(k) \leq \sum_i Mw_i = M \sum_i w_i = MW_{sum}. \quad (3.10)$$

Therefore, a maximum of WM_{sum} packets may be served before the scheduler begins to serve the newly active flow. The statement of the theorem is then readily proven and the access delay is bounded by

$$D \leq \frac{\sum_i Sent_i(k)}{C} \leq \frac{MW_{sum}}{C}. \quad (3.11)$$

3.5.4 Isolation

VSQ offers isolation among different flows by providing separate source queues. The services are isolated from each other. Ill-behaving sources cannot abuse the network resources. Excessive packets from such sources experience long transit time through the ring but they do not affect the packets from the well-behaving sources. In RPR, on the other hand, since there is no separation of sources, a well-behaving source may be starved by ill-behaving ones. The isolation in VSQ provides a strong incentive for ill-behaving sources to improve their behavior. The simulation results on the isolation property of VSQ are shown in Chapter 4.

3.6 Recommendations for setting VSQ parameters

Because of the requirements for fair rate estimation and feedback mechanism, RPR needs to set more than twenty heuristic parameters, such as, weights, rate rump up/down parameters, buffer thresholds, control intervals, and low pass filter coefficients, etc. The setting of these parameters has major impact on the performance metrics such as

convergence time [35][37] In contrast, VSQ does not require fair rate estimation and feedback fairness control, hence, it only requires setting of three parameters: weight, *VSQHighThreshold* and *VSQLowThreshold*. The guidelines for setting these parameters are described in the following.

3.6.1 Weight

As part of the scheduler setting, a weight is used in distributing the FE traffic over a congested link. A node with higher weight can get larger bandwidth share in a congested link. We model a flow as a sequence of packets from the same source node over a series of nodes. The weight of the flow is the same as the weight of the source node.

Since every node has a minimum bandwidth requirement for its fairness eligible traffic, the weights can be set based on the minimum bandwidth requirement of a node. For simplicity of explanation, we consider that VSQ scheme is applied to a fixed networking configuration. We assume during the initialization phase, each node i of an N -node ring requests a minimum bandwidth, r_i ($1 \leq i \leq N$). Let r_{min} represent the lowest value of r_i 's and C represent the unreserved ring bandwidth. Since all the FE traffic share the unreserved ring bandwidth, a necessary constraint is that the sum of the r_i 's should not exceed the unreserved link capacity:

$$\sum_{i=1}^N r_i \leq C. \quad (3.12)$$

To allow each flow to receive the service proportional to its minimum bandwidth requirement, VSQ scheme assigns a weight to each flow. The weight assigned to flow i , w_i , is given by:

$$w_i = \frac{r_i}{r_{\min}}. \quad (3.13)$$

We demonstrate this weight assignment procedure in the following example.

Example 3-4: Consider a ring with $N=6$ and link capacity of 622Mbps. Assume that 72 Mbps of the bandwidth is reserved for high priority traffic, the remaining bandwidth of 550 Mbps is shared by 6 nodes with the matrix of the minimum bandwidth requirements given as

$$r_1 = r_2 = r_3 = r_4 = 50 \text{ Mbps},$$

$$r_5 = 100 \text{ Mbps},$$

$$r_6 = 200 \text{ Mbps}.$$

The requirements from the six nodes satisfy equation (3.12), i.e.,

$$\sum_{i=1}^6 r_i \leq 550 \text{ Mbps},$$

and we have

$$r_{\min} = 50 \text{ Mbps}.$$

The weights for these six nodes can then be assigned using (3.13) as

$$w_1 = w_2 = w_3 = w_4 = 1,$$

$$w_5 = 2,$$

$$w_6 = 4.$$

3.6.2 Buffer Thresholds

The queue thresholds, $VSQHighThreshold$ and $VSQLowThreshold$, are set based on two factors: 1) delay requirement and 2) buffer budget. The general equation for buffer size is given as:

$$Buffer\ size = delay * buffer\ growth\ rate \quad (3.14)$$

We can view the transit buffer as three parts using the $VSQHighThreshold$ and $VSQLowThreshold$ as shown in Figure 3-8.

Figure 3-8: Illustration of VSQ transit buffer usage

The first part is used to achieve maximum utilization of the ring bandwidth. This is because there is a one-hop-delay for ON signal to take into effect. During this period of time, the transit buffer occupancy should not fall to zero. The buffer growth rate is the line rate minus the reserved rate, R_{re} . Thus, we have:

$$VSQLowThreshold > one-hop-delay * (line\ rate - R_{re})$$

The third part of the transit buffer is used to absorb packets during the one-hop-delay for the OFF signal to take into effect. Therefore, $VSQHighThreshold$ should be at least the buffer size, $sizeVSQ$, minus the one-hop-delay effect:

$$VSQHighThreshold \leq sizeVSQ - one-hop-delay * (line\ rate - R_{re})$$

The second part of the transit buffer represents the difference between the two thresholds and decides on the frequency of oscillations between ON-OFF signals. The difference between the low and high thresholds can be either determined empirically or left to an administrator to tune for protocol performance. Our experience is that the feedback signals should be sent when the buffering space is exhausted about halfway; this is not based on extensive experimentation but appears to be a reasonable engineering decision [65]. There is a tradeoff between the response time and control messages depending on the threshold value. Higher threshold means larger available buffer, slower response, and less control messages; lower threshold means smaller available buffer, faster response, and more control messages. Dynamic threshold setting [66][67][68] based on the traffic demand is also possible in VSQ at the expense of complexity, however, we have not found this to be necessary.

The third part of the transit buffer is used to hold the incoming transit packets until the backpressure message takes into effect. When the size of source queue(i) exceeds *VSQHighThreshold*, it means that the source queue(i) builds up and there is a possibility of buffer overflow. Hence, a backpressure message needs to be employed to inform the neighboring upstream node to stop serving source queue(i). With the OHB control scheme, there is only a small one-hop-delay from the time the backpressure message is sent to the immediate upstream neighbor to the time the neighbor stop sending to this node. As a result, the third part of the buffer could be quite small. For example, for a link of 10 km, the one-hop-delay is 0.05ms, we have:

$$Buffer\ size_part3 = 0.05ms * 622Mbps = 3.8KB$$

The thresholds are normally set as the percentage of the buffer size. Let B denote the total buffer size, N denote the number of nodes, d for the one-hop-delay and C to be the link rate. For simplicity, we assume all nodes have the same weight and all source queues have the same threshold. The thresholds can then be set as the ratio of the buffer size to ensure that there is no packet loss, shown in (3.15) and (3.16). An illustrative example is also given to show how to set the thresholds.

$$VSQHighThreshold = \frac{(B/N) - d \times (C/N)}{2(B/N)} = \frac{B - d \times C}{2B} \quad (3.15)$$

$$VSQLowThreshold = \frac{(B/N) - d \times (C/N)}{4(B/N)} = \frac{B - d \times C}{4B} \quad (3.16)$$

Example 3-5: Consider a ring with link rate of $C = 622Mbps$, one-hop-delay of $d = 0.1 ms$, and buffer size of $B = 200 Kbyte$. We can set the thresholds as:

$$VSQHighThreshold = \frac{2 \times 10^5 - 0.0001 \times 622 \times 10^6}{2 \times 2 \times 10^5} \approx 70\%$$

$$VSQLowThreshold = \frac{2 \times 10^5 - 0.0001 \times 622 \times 10^6}{4 \times 2 \times 10^5} \approx 35\%$$

3.7 Summary

We identify that the *root cause* of the RPR unfairness problem lies in the priority scheduling of the MAC. Priority scheduling is unable to achieve simultaneous fairness and high utilization for fairness eligible traffic, and hence, is not suitable for FE traffic.

To address the root cause, we propose a new MAC protocol, called VSQ, for packet ring networks. VSQ offers traffic isolation by providing a virtual queue for each source. It uses a fair scheduler to serve these source-based queues, and therefore, ensures fairness

and maximizes ring utilization without the need for fair rate estimation and feedback control.

We propose OHB as an efficient flow control scheme to provide a lossless transit path for VSQ. OHB exploits the source information to acquire the knowledge of the traffic rates of the upstream nodes and signal them when buffer overflow is imminent. OHB works very effectively as it can evenly spread the buffer space along the path between the congested node and the source. On the contrary, RPR lets the buffer space concentrated at the congested node. As a result, VSQ requires less transit buffer size compared to RPR.

In summary, VSQ enjoys the following advantages with respect to the MAC design goals:

- 1) Fairness guarantee: Fairness is ensured among competing sources as the WRR scheduler fairly serves the queues on a weighted basis.
- 2) Throughput maximization: Bandwidth is shared perfectly because the scheduler is work-conserving. Every link of the ring can be kept at full utilization at all times provided that there is enough data to transmit.
- 3) Low access delay: Because VSQ does not rely on feedback rate control, every node can access its fair share with a low bounded access delay.
- 4) Allow all sources to have a *fast-start*: VSQ does not conduct rate control at the ingress point. Instead, it allows every node send as much FE traffic as possible at the ingress point. Then, it serves the traffic in a fair and best-effort manner along the transit path resulting in high utilization while guaranteeing fairness.
- 5) Lossless transit paths: By exploiting source information, VSQ enables a simple and effective OHB flow control mechanism to ensure a lossless transit path.

- 6) Isolation: VSQ isolates the traffic of various sources. Upon congestion, only those nodes that send more than their fair share suffer delay.
- 7) Simplicity: There is no need for fair rate estimation and feedback to ensure fairness. In comparison with RPR, VSQ requires fewer parameters to set and it also sends fewer control messages.

The requirements for VSQ are:

- 1) Virtual queues in the transit buffer. Since the queues are virtual queues, they can be implemented as counters or link list (as in DVSR). The number of node in a metro packet ring is limited, e.g., RPR standard limits the number of nodes on a ring to 256 [15], thus, the number of virtual queues per node is limited to 256. In addition, the complexity added by the virtual source queues in VSQ is offset by the elimination of VDQs that are required in RPR.
- 2) Fair scheduler: WRR scheduler implementation is slightly more complex than the priority scheduler used in RPR.

Figure 3-9: MAC layer comparison of VSQ, RPR and OSI reference model

Figure 3-9 illustrates the comparison of VSQ layers to RPR layers. RPR MAC data path is based on BIR using priority scheduling between the transit traffic and node traffic. VSQ MAC utilizes fair scheduler to serve the virtual source queues. We can also see that the control layer of VSQ is simpler since the overlay fairness control module is not required due to the inherent fairness property of VSQ. Other parts of MAC control layer such as OAM, and topology & protection are the same for both VSQ and RPR.

In the next chapter, we show the performance comparison between VSQ and RPR through both analysis and simulations.

CHAPTER 4 Performance Evaluation of VSQ

4.1 *Introduction*

The previous chapter presented the design and analysis of the VSQ MAC and the OHB flow control scheme. This chapter applies both mathematical analysis and extensive simulations to study the performance of VSQ and compare the results with those of RPR standard. The performance results also justify the claims made in Chapter 3.

We first present the evaluation methodology including analytical modeling, performance metrics, and simulation tools in Section 4.2. We then present, in Section 4.3, the results on the compliancy of VSQ with packet ring MAC design goals such as: source-based weighted fairness, spatial reuse, maximization of link utilization and isolation between sources. Finally, we demonstrate the performance superiority of VSQ in terms of throughput, fairness convergence time, delay and transit buffer size compared with those of the RPR standard (Section 4.4 - 4.8).

4.2 *Evaluation Methodology*

4.2.1 *Analytical Modeling*

Analytical modeling is used to study the average access delay and end-to-end delay for both VSQ and RPR. Although some system assumptions are made to make mathematical analysis possible, the analytical models can provide good insights into the system performance.

The VSQ analytical results on both average access delay and end-to-end delay are based on the Cyclic Polling model with k -limited Service [70][71][72] whereas the RPR

analytical results for delay are based on the head-of-line non-preemptive queuing analysis [73][74]. In both cases, certain assumptions are made for mathematical tractability, such as Poisson traffic arrivals, uniform traffic pattern and independent service times.

4.2.2 Simulation Tool

The simulation tool we use in this thesis study is OPNETTM Modeler (Version 9.1) that is a commercial discrete event simulator for network research [75]. It provides substantial support for MAC, traffic generation and performance analysis, making it ideal for evaluating MAC schemes.

4.2.3 Performance Metrics

The two main performance metrics used in MAC protocol evaluation are throughput and access delay. Throughput refers to the number of bits sent from the source to the destination and correctly received, over certain period of time. Access delay is the time taken by a packet to get on the ring. Sources with FE traffic are throughput sensitive sources and prefer large throughputs. Access delay is a measure to show that how fair a MAC is. The less access delay means the faster a node can access its share of ring bandwidth.

In our simulations, we measure throughputs over simulation intervals. The time scale over which network state changes is determined to be much larger than the time scale of network dynamics. Access delays are measured at each IB. It is assumed that there are no packet losses in transmission due to bit corruption or noisy line.

4.2.4 Simulation Setup

Table 4-1 lists the characteristics of the packet ring that we consider in our study.

Table 4-2 lists the parameters used in the simulation of RPR scheme.

Table 4-1: Characteristics of the test network

Nodes	10
Ring size	100km
Propagation delay per link	50 μ s
Speed	622Mbps
Packet size	500 Bytes
Transit buffer size	256 KB

Table 4-2: RPR parameters

Low pass filter coefficient	0.5, 0.95
Transit buffer threshold	0.5 (128KB)
Control interval	100 μ s
Advertisement interval	100 μ s

We consider ten benchmark scenarios to evaluate the performance of the VSQ and RPR MAC schemes. The brief descriptions of these scenarios are given in Table 4-3. Scenarios 1 to 4 study the compliancy of VSQ with MAC requirements such as weighted fairness, spatial reuse, dynamic responsiveness and isolation property. Scenarios 4 to 10 compare the performance of VSQ and RPR in terms of throughput, fairness convergence time, average delay and transit buffer occupancy. We focus on FE traffic, i.e., LP traffic, in all scenarios except scenarios 9 and 10 in which we include HP traffic in order to study the transit buffer occupancy.

Table 4-3: Simulation Scenarios and Results

Scenario #	Description	Results
1: Weighted fairness	Three greedy flows with different weights sharing one link	VSQ allows each flow to receive a weighed share of the link.
2: Spatial reuse	Two congested domains	VSQ provides full utilization of both domains.
3: Dynamic responsiveness	Some sources turn off	VSQ can reclaim the unused bandwidth very quickly.
4: Isolation	Sources with heterogeneous demand	VSQ provides isolation between well-behaving nodes and ill-behaving node
5: Throughput comparison	Unbalanced traffic	VSQ offers higher throughput than RPR does.
6: Convergence comparison	Balanced traffic	VSQ offers less convergence time than RPR and DVSR does.
7: Average delay comparison	Poisson traffic model	VSQ offers less access delay, same or less end-to-end delay compared to RPR.
8: Average delay comparison	Pareto On/Off traffic model	VSQ offers less access delay, same or less end-to-end delay compared to RPR.
9: Transit buffer occupancy	HP traffic causing LP transit buffer growth.	VSQ has less transit buffer occupancy than RPR.
10. Transit buffer occupancy variance	Increase the distance between the congested node and the source node for RPR.	When the distance between the congested node and the source node increases, the buffer occupancy variance increases with RPR but not with VSQ

The values of the line speeds, delays and buffer sizes in the scenarios are chosen to accentuate the differences between the MAC schemes. In many scenarios, we study the traffic sources with demand of line rate, i.e., greedy sources, thus, resource contention always occurs at the output link of down stream nodes where fairness algorithms are activated.

4.2.5 Format of the Results

We present the simulation results as tables and as graphs of the target performance metric versus simulation time or system load. Examples of such graphs include throughput versus simulation time and packet delay versus system load.

Both throughput and delays are averaged over certain interval. Since simulation experiments typically run for many intervals, it is possible to calculate the standard deviation of the means over each interval. If each interval is long enough, then the distribution of the interval means of a metric can be approximated by a Gaussian distribution. Thus, 95% confidence intervals for the mean of means can be calculated as thrice the standard deviation on each side of the mean of means. This computation assumes that an interval is long enough so that simulation dynamics are averaged out. One way to achieve this is by linking the length of a simulation run to the confidence in the metrics [76][77]. However, we simply choose measurement intervals that are conservative enough that the standard deviation of the mean is fairly small, so that the confidence in the mean of means is high. Further, we ignore the mean over the first interval to eliminate transient behavior of the initial phase.

4.3 Properties of VSQ

4.3.1 Weighted Fairness

As mentioned earlier, source-based weighted fairness is a basic goal defined by the RPR standard [15]. In this experiment, we assign different weight to each source and demonstrate that VSQ can achieve the weighted fairness.

As shown in Figure 4-1, source 1 has weight 1, source 2 has weight 2, and source 3 has weight 4. There are three flows namely flow 1-5, 2-5, and 3-5 that share link 4-5. The demand of each flow is the link rate. Figure 4-2 shows that all three flows are able to obtain their fair bandwidth share of the bottleneck link 4-5 based on their assigned weights, i.e., 90 Mbps for flow 1-5 with weight of 1, 180 Mbps for flow 2-5 with weight of 2 and 360Mbps for flow 3-5 with weight =4.

Figure 4-1: Weighted fairness setup

Figure 4-2: Weighted fairness results

4.3.2 Spatial Reuse

Spatial reuse deals with the simultaneous transmission over non-overlapping segments of the ring. Spatial reuse is one of the basic requirements of the MAC scheme for packet rings.

To demonstrate the spatial reuse capability of VSQ, we use the configuration with two congestion domains (CDs) as shown in Figure 4-3. The first congestion domain is from node 1 to node 4 with the congested link being the link 3-4. The flows from node 2 and node 3 demand 622 Mbps and the flow from node 1 demands 100 Mbps. The second congestion domain is from node 5 to node 8 with the congested link 7-8. There are three flows sharing link 7-8, each flow demands the full link rate of 622 Mbps. The two congestion domains do not overlap with each other. The objective is to achieve source based fairness, with a fair rate of 260 Mbps for flow 2-4 and flow 3-4 in the first congestion domain and a fair rate of 207 Mbps for all three flows in the second congestion domain.

The simulation results of this scenario are shown in Figure 4-4. As expected, the throughput of the flow 2-4 and flow 3-4 in the first congestion domain converges to the fair rate of 260 Mbps very rapidly and the throughput of flows 5-8, 6-8, and 7-8 in the second congestion domain converge to the fair rate of 207 Mbps. The results not only indicate 100% utilization for both links 3-4 and 7-8 but also demonstrate the capacity of VSQ in achieving spatial reuse.

Figure 4-3: Scenario with two bottleneck links: link 3-4 and link 7-8.

Figure 4-4: Throughput of 6 flows from two congestion domains.

4.3.3 Dynamic Responsiveness

In this section, we demonstrate the responsiveness of VSQ scheme to the dynamic changes in traffic demand. In our simulation, we consider bursty ON/OFF traffic pattern.

Figure 4-5. Packing lot scenario with dynamic traffic

Figure 4-6. Results showing fast reclamation of available bandwidth.

Figure 4-5 shows the simulation scenario in which four flows start and stop at different points of time. Flow 4-5 starts at 0 second, flow 3-5 starts at 0.1 second, flow 2-5 starts at 0.2 second, and flow 1-5 starts at 0.3 second and stops at 0.4 second. All the flows are greedy as they demand the full link rate. We focus on the throughput at the bottleneck link 4-5.

With the addition of each new flow, VSQ re-allocates the bandwidth of link 4-5 among the contending flows. As we can see from Figure 4-6 that the fair rate converges very fast. At time 0.1 second (T1), the two contending flows over link 4-5 are flow 4-5 and flow 3-5, both converge to the fair rate of 311 Mbps. At time 0.2 seconds (T2), the three contending flows of 4-5, 3-5 and 2-5, converge to the fair rate of 207 Mbps. Similarly, at time 0.3 seconds (T3), the four flows of 4-5, 3-5, 2-5, and 1-5 converge to the fair rate of 155 Mbps.

When flow 1-5 stops at 0.4 second (T4), the remaining three flows (flow 2-5, flow 3-5, flow 4-5) reclaim the bandwidth left by flow 1-5 immediately, and the fair rate ramps up from 155 Mbps to 207 Mbps. Once flow 2-5 stops at 0.7 second (T5), flow 3-5 and flow 4-5 reclaim the bandwidth left by flow 2-5, and the fair rate increases from 207 Mbps to 311 Mbps with minimal delay. These observations indicate that VSQ scheme maximizes the bandwidth utilization by reclaiming the released bandwidth from sources that become idle.

4.3.4 Isolation of Ill-behaving Sources

VSQ provides protection for traffic flows, i.e., well-behaving flows are isolated from ill-behaving ones. We demonstrate this property through the scenario shown in Figure 4-

7 where three flows, 5-8, 6-8 and 7-8 share the link 7-8. Flow 6-8 demands 207 Mbps which equals its minimum guaranteed fair rate, hence, it experiences minimum access delay and queuing delay. On the other hand, flow 5-8 and flow 7-8 require 622 Mbps which is much more than their fair share, hence, both flows suffer excessive delay. Figure 4-8 shows the access delay comparison between the well-behaving flow 6-8 and the ill-behaving flow 5-8.

Figure 4-7: Scenario to show isolation

Figure 4-8: Results showing isolation property

While in RPR, there is no guarantee for fair access. Although node 6 only demands its fair share, it still suffers large access delay because it can only access its fair share of ring bandwidth after the fair rate feedback control takes into effect, i.e., after node 5 reduces its sending rate.

4.4 Throughput Comparison

Throughput is one of the important performance objectives of all MAC schemes. In this section, we compare the throughput of RPR and VSQ for a given simulation scenario.

We use the “unbalanced” scenario described by Figure 2-8 and redraw it in Figure 4-9. For this scenario, the rate of flow 1-3 oscillates between the low rate 5 Mbps (which provide fair sharing) and high rate 617 Mbps (which allows throughput maximization) with RPR-AM as shown in Figure 2-9. However, with VSQ, the rate of flow 1-3 starts at the full link rate of 622 Mbps and within 1 round-robin cycle time, reduces its rate to 617 Mbps, which is the optimal fair rate.

Bandwidth oscillation leads to throughput loss. Figure 4-10 plots the link utilization of link 2-3 for both RPR-AM and VSQ. The throughput plot which we omit here has a profile similar to that of the link utilization plot. As we can see, the link utilization of RPR-AM is at 70% and VSQ is at 100%, which means RPR-AM suffers 30% throughput loss while VSQ has no throughput loss.

Figure 4-9: Unbalanced traffic scenario with two flows

Figure 4-10: Link utilization comparison of RPR and VSQ

4.5 Fairness Convergence Time Comparison

Fairness convergence time refers to the amount of time it takes to converge to fair rate. It is different for every fairness control algorithm. The faster the fair rate converges,

the better the fairness scheme is. Hence, we adopt the fairness convergence time as one of the performance metrics to evaluate fairness control schemes.

Figure 4-11 shows the simulation scenario to study the convergence time. The link rate is set to be 622Mbps. Source nodes 1, 2, 3, and 4 start sending 248 Mbps (40% of the link rate) to node 5 at times 0 second, 0.1 second, 0.2 second and 0.3 second, respectively. In this example, the output link of node 4, link 4-5, is the bottleneck and it is shared by sources 1, 2, 3 and 4 with equal weights.

Figure 4-11: Scenario to study convergence time

Figure 4-12 and 4-13 shows the convergence of RPR and VSQ respectively. We see that RPR presents some bandwidth oscillations before converging to the fair rate. On the other hand, VSQ converges to the fair rate without any bandwidth oscillation: for the time interval $[0.2, 0.3)$ second, source 1 and 2, 3 share the output link of node 4 and each receives the fair rate of $(622/3) = 207$ Mbps; for the time interval of $[0.3, 0.5)$ second, source 1, 2, 3 and 4 share the link, each with the fair rate of $(622/4) = 155$ Mbps.

Figure 4-12: Convergence of RPR

Figure 4-13: Convergence of VSQ

Figure 4-14 compares the fair rate convergence time of VSQ with the results in [27]. The convergence times are 50msec for RPR-AM, 18 msec for RPR-CM, 2 msec for DVSR and 0.1msec for VSQ. The convergence time in both RPR and DVSR is a function of the ring Round Trip time (RTT) [27],[36]. For example, if the RTT increases by a factor of 10 from 1 msec to 10 msec, DVSR takes approximately 22 msec to converge [27]. The reason is that the feedback delay depends on the RTT. On the other hand, the RTT has no impact on the VSQ convergence time because VSQ requires neither fair rate estimation nor feedback control to achieve fairness [78].

Figure 4-14: Fair rate convergence time comparison

4.6 Access Delay Bound Analysis

Recall that RPR MAC is based on BIR, in which the transit traffic has priority over the local add-on traffic. Therefore, all add-on traffic may experience ring access delay [24]. The access delays are essentially affected by how long a continuous traffic stream is received from the upstream nodes.

To reduce the access delay, fairness algorithm is designed to control the access of the ring bandwidth for all nodes during periods of congestion. The less access delay, the better the MAC design is. Thus, access delay is a good performance metrics to evaluate the effectiveness of a fairness control scheme. *Access delay bound* is defined as the access delay at the worst case. In the following subsections, we derive and compare the access delay bound of VSQ with that of RPR.

4.6.1 Access Delay Bound for VSQ

We re-iterate our derivation of the access delay bound for VSQ in section 3.4.3. For an N -node ring, there are maximum $(N-1)$ virtual source queues at each node. Unlike RPR, there is no need for rate feedback control in VSQ. Hence, the upper-bound for access delay is the maximum waiting time for a packet at the head of the local add-on queue, which is the time that the scheduler needs to serve all other $(N-1)$ virtual source queues. We assume that the switch over time between the queues is negligible, the weight for each source is equal to 1 and the service rate for LP traffic, i.e., unreserved rate, is C . The worst case happens when all $(N-1)$ queues are not empty and the packets are at the maximum size of M bits. As a result, the access delay upper bound, VSQ_{bound} , is given as:

$$VSQ_{bound} = M \cdot (N - 1) / C. \quad (4.1)$$

Figure 4-15 illustrates the worst case scenario in which a packet at queue N (local traffic queue)¹³, can access the ring after the scheduler has served a packet from all other $(N-1)$ non-empty queues.

¹³ Without loss of generality, we consider the N^{th} subqueue as the local add-on queue.

Figure 4-15: Illustration of access delay bound derivation in VSQ.

4.6.2 Access Delay Bound for RPR

The RPR MAC uses a priority scheduler that assumes the traffic in the transit buffer has higher priority over the traffic in the node buffer. A starved node can only access the ring once the rate feedback control has taken into effect. As a result, the access delay bound equals to the *responseTime* defined as the maximum amount of time from the node becomes congested until the node stops receiving traffic that causes congestion.

We first give a graphical presentation of the *responseTime* and we then derive the RPR access delay bound by discussing its components.

1) Graphical representation of *responseTime*

Figure 4-16 illustrates the time diagram of the RPR feedback control scheme where there is a congestion domain consists of a head, tail and intermediate node. Time increases down the vertical axis, and each axis represents a node along the path of a flow. Lines from the congestion domain head to the tail show the transmission of a feedback

control message. The parallelograms represent two kinds of delays: the vertical sides represent the delay at a node. This delay includes the transmission delay, which is the packet size divided by the line capacity; and processing delay at a node, which is the time waiting for advertisement interval, and applying to the rate controller, etc. The slope of the longer sides is proportional to the propagation delay. Once a node receives the fair rate, it needs first drain its transit buffer before the new fair rate can take into effect.

Figure 4-16: Time diagram for fair rate control cycle: $responseTime \sim (t_6 - t_0)$

Using this diagram, $responseTime$ is shown to be $t_6 - t_0$ where t_0 is the time that the congestion head node wants to add traffic to the ring and t_6 is the time that this action is actually performed. $responseTime$ can be written as:

$$responseTime = t_6 - t_0 = \sum_{i=1}^6 (t_i - t_{i-1})$$

where

$t_1 - t_0$: time to detect congestion and generate rate control message,

$t_2 - t_1$: one-hop propagation delay. Only one intermediate node is shown here for illustration purpose only. When there are multiple intermediate nodes in the congestion span, this delay will be multiple hop delay,

$t_3 - t_2$: nodal processing time and forwarding delay,

$t_4 - t_3$: one-hop propagation delay,

$t_5 - t_4$: nodal processing time and forwarding delay: time to drain the transit buffer before the new fair rate can be applied,

$t_6 - t_5$: propagation delay from the tail node to the head node.

2) Derivation of the bound

From previous discussion, *responseTime* can be generalized into the summation of three parts:

$$responseTime = agingInterval + \max FRTT + \max FRED \quad (4.2)$$

The first part in (4.2), *agingInterval*, is the maximum amount of time for the fairness algorithm to detect the congestion. The RPR standard specifies the default value for *agingInterval* to be $100\mu s$ for link rate above 622Mbps and $400\mu s$ for link rate below 622Mbps.

The second part, *maxFRTT* (Maximum Fairness Round Trip Time), is assumed to be the sum of the round trip time for propagating the fairness message around the ring and the nodal delay of fairness message. Hence, *maxFRTT* can be estimated as follows [15],

$$\max FRTT = 2 * 5\mu s / km * L + advertisingInterval * N \quad (4.3)$$

where L is the ring length (or circumference) in kilometer (km), link propagation delay is $5\mu s/km$ based on $2*10^8$ m/sec speed of light on fiber, N represents the number of nodes,

and the *advertisingInterval* represents the maximum delay of the fairness message at a node, i.e., the fairness message waits maximum *advertisingInterval* to be forwarded to next hop¹⁴.

The last part, *maxFRED* (Maximum Fair Rate Enforcement Delay), is the time required for the fair rate to be taken into effect at all ring nodes. It is different for 1-TB and 2-TB design as explained in the following.

- **MaxFRED in 1-TB design:** Each RPR node is equipped with a token bucket to shape the add-on traffic to the fair rate. The size of the bucket decides on the amount of burst each node can send. Once a node receives the fair rate, it changes its token generation rate to this rate. A new rate will not be effective until the buckets are cleared from the tokens generated based on the previous rate. This may create a large burst concatenated by the burst of individual buckets. The larger the burst from upstream nodes, the longer the access delay for the downstream node.

Figure 4-17 shows the bursts generated by upstream nodes are concatenated and formed a longer burst when they reach downstream nodes. At worst case scenario, a congested downstream node can observe the aggregation of the longest burst of all upstream nodes [79]. Note that the transit buffer of 1-TB mode is very small and we neglect its impact on the bursts. Let us consider an N -node ring in which each node i is regulated by a token bucket with parameter $(\sigma_i$,

¹⁴ This is the worst case assumption that the congestion domain extends for the entire ring and each hop delay lasts for the maximum amount of time which is the *advertisementInterval*.

ρ_i). The token bucket credit limits of all the upstream nodes should be cleared

before the fair rate can take into effect, thus, $maxFRED$ is given as:

$$maxFRED_{(1-TB)} = \frac{\sum_{i=1}^{N-1} \sigma_i}{C - \sum_{i=1}^{N-1} \rho_i}, \quad (4.4)$$

where $\sum_{i=1}^{N-1} \sigma_i$ is the maximum number of tokens that the $(N-1)$ upstream nodes

can accumulate, C is the unreserved rate for LP traffic, and $\sum_{i=1}^{N-1} \rho_i$ is the total

token replenishing rate of the $(N-1)$ upstream nodes and it also represents the total transit traffic rate to node N .

Figure 4-17: Illustration of a congestion domain and burst concatenation

- **MaxFRED in 2-TB design:** RPR 2-TB design defines a large LP transit buffer which affects the access delay of a downstream node. Before the effect of a rate change at the congestion domain tail node can be observed by the head node, all LP

transit buffers between these two nodes must be emptied. By definition, *stqFullThreshold* is the highest level of LP transit buffers (or Secondary Transit Queues, STQs). At the worst case scenario, shown in Figure 4-18, all upstream LP transit buffers are at this level when they receive the new fair rate. Thus, they have to drain *stqFullThreshold* worth of bytes before they can apply the new fair rate. Note that since *stqFullThreshold* is much greater than the token bucket size, we neglect the effect of the burst generated by the bucket size. With the service rate of C for LP transit buffer, the value of $\max FRED^{15}$ for an N -node ring can be estimated as [15]:

$$\max FRED_{(2-TB)} = stqFullThreshold \cdot (N - 1) / C. \quad (4.5)$$

Figure 4-18: Illustration of *maxFRED* derivation for RPR-2TB

¹⁵ *maxFRED* is under the term *queueDrainTime* in the RPR standard. It represents the maximum amount of time that could be incurred waiting for the upstream STQs to be drained. It is assumed that all upstream STQs filled up to their *stqFullThreshold* at the time they receive the new fair rate, and therefore have to drain *stqFullThreshold* worth of bytes before they fully back off to the amount requested by the fair message.

With the above discussions, the access delay bound for RPR can be obtained through the following two equations.

$$RPR_{bound(1-TB)} = agingInterval + \max FRTT + \max FRED_{(1-TB)} \quad (4.6)$$

$$RPR_{bound(2-TB)} = agingInterval + \max FRTT + \max FRED_{(2-TB)} \quad (4.7)$$

where *agingInterval* is a set value based on the link rate, *maxFRTT* is calculated by (4.3) and *maxFRED* is obtained by (4.4) and (4.5) for 1-TB and 2-TB design, respectively.

4.6.3 Comparison

In this section, we compare the access delay bound of VSQ and RPR and show that VSQ has smaller access delay bound. First we compare the VSQ bound with the bound of RPR 1-TB implementation.

$$\frac{RPR_{bound(1-TB)}}{VSQ_{bound}} = \frac{agingInterval + \max FRTT + \max FRED_{(1-TB)}}{M \cdot (N-1) / C} \quad (4.8)$$

Since

$$\begin{aligned} agingInterval &> 0, \\ \max FRTT &> 0, \end{aligned}$$

and

$$\max FRED_{(1-TB)} = \frac{\sum_{i=1}^{N-1} \sigma_i}{C - \sum_{i=1}^{N-1} \rho_i},$$

we then have:

$$\frac{RPR_{bound(1-TB)}}{VSQ_{bound}} > \frac{\frac{\sum_{i=1}^{N-1} \sigma_i}{C - \sum_{i=1}^{N-1} \rho_i}}{M \cdot (N-1) / C}. \quad (4.9)$$

Since

$$\frac{\sum_{i=1}^{N-1} \sigma_i}{C - \sum_{i=1}^{N-1} \rho_i} \cdot C > \sum_{i=1}^{N-1} \sigma_i,$$

then

$$\frac{RPR_{bound(1-TB)}}{VSQ_{bound}} > \frac{\sum_{i=1}^{N-1} \sigma_i}{M \cdot (N-1)} \quad (4.10)$$

The RPR standard defines the maximum value of the token bucket size σ_i to be M bits, hence,

$$\sum_{i=1}^{N-1} \sigma_i > M \cdot (N-1).$$

By applying the above inequality into (4.10) we have:

$$\frac{RPR_{bound(1-TB)}}{VSQ_{bound}} > 1 \quad (4.11)$$

Similarly, we compare the access delay bound of VSQ with that of RPR 2-TB implementation as:

$$\frac{RPR_{bound(2-TB)}}{VSQ_{bound}} = \frac{agingInterval + \max FRTT + \max FRED_{(2-TB)}}{M \cdot (N-1) / C} \quad (4.12)$$

Since

$$\begin{aligned} agingInterval &> 0, \\ \max FRTT &> 0, \end{aligned}$$

and

$$\max FRED_{(2-TB)} = stqFullThreshold \cdot (N-1) / C,$$

then,

$$\frac{RPR_{bound(2-TB)}}{VSQ_{bound}} > \frac{stqFullThreshold \cdot (N-1) / C}{M \cdot (N-1) / C} = \frac{stqFullThreshold}{M} \quad (4.13)$$

The RPR standard specifies the range for *stqFullThreshold* as $(4M, stqSize-M)$. By applying the minimum value of $4M$ to (4.13), we obtain:

$$\frac{RPR_{bound(2-TB)}}{VSQ_{bound}} > 4 \quad (4.14)$$

Despite the conservative consideration in deriving the relation between the bounds of RPR and VSQ, we can conclude that a VSQ node experiences less access delay bound than an RPR node. The reason is that the add-on traffic of a VSQ node has immediate access to its fair share of ring bandwidth and it does not wait for the upstream nodes to slow down in order to access the ring. While in RPR, since the ring traffic assumes higher priority, the local traffic can only be added to the ring when the upstream nodes slow down their traffic through the application of a rate feedback control scheme.

4.7 Average Delay Analysis

Although worst-case access delay can be obtained from the above access delay bound, both average access delay and end-to-end delay are of interest for metro service provisioning. We define these quantities before we discuss the details of analysis for both RPR and VSQ.

Definition 4-1: *Average access delay* is the average waiting time that a packet experiences from the time it arrives at a node to the time it gets on the ring.

Definition 4-2: *Average end-to-end delay* is the average waiting time that a packet experiences from the time it arrives at a node to the time it is delivered to its destination node.

It is very difficult, if not impossible, to derive the exact results for the above two performance metrics. However, we can provide approximate results for these quantities using certain assumptions that allow mathematical tractability.

4.7.1 System Assumptions

We consider the following assumptions for both VSQ and RPR delay analysis:

- No overlapped transmission: a packet does not leave a node until its transmission is over, hence, the transmission times are additive.
- The capacity of all buffers is infinite.
- The interarrival times and service times are independent, thus, we can use independent sub-models for each node.
- The add-on traffic to each node is independent and follows Poisson process with mean arrival rate of λ .
- All add-on traffic have the same distribution of packet lengths, with first and second moments $\bar{L}$ and $\bar{L}^2$ respectively. If the system service rate is R , then the packet service time is $x=L/R$ and has first and second moment of $\bar{x}$ and $x^{(2)}$, respectively.
- Symmetric traffic pattern: Each node sends uniformly to all other nodes, i.e., node i transmits to node $(i + j) \bmod N$ (where $j = 1, 2 \dots N - 1$) with the same probability of $1/(N-1)$.
- Link propagation delay has been excluded from delay analysis of both VSQ and RPR.

4.7.2 Average Delay Analysis for VSQ

Each VSQ node is viewed as a system with multiple queues served with a WRR scheduler. This system can be modeled as the *Cyclic Polling Model with k-limited Service*, in which a server cycle through multiple queues serving up to a maximum of k packets (i.e., service limits $\leq k$) from each queue [51]. This queuing problem has been a challenging area and a lot of research has been carried out over the past decade. Much of the research has focused on how to accurately estimate the queuing delays. Unfortunately, except for a few special cases [70], there are no exact solutions for expected delays experienced by the packets in each queue for most of the general cases.

The good news is that many other approximations have been proposed to estimate the average delays based on the so-called *Pseudo-Conservation Law (PCL)* [71]. The PCL provides a closed-form solution for a specific weighted sum of the mean waiting times. One of the advantages of the PCL is that it is useful in developing simple and reasonably accurate approximation for the individual mean waiting times. Fuhrmann and Wang [72] have provided such an approximation for k -limited service. This approximation formula is relatively simple and captures the major factors important for efficient operation. Authors of [80] have verified the accuracy of this approximation and applied it in their studies. After examining the accuracy of this delay approximation, we use it in our studies.

Let us first list the notations that are used in the analysis of VSQ:

$Q_i(j)$: the j^{th} virtual source queue of node i . ($1 \leq i \leq N$, $1 \leq j \leq N$),

$w_i(j)$: weight of $Q_i(j)$, i.e., service limit of $Q_i(j)$.

$\lambda_i(j)$: packet arrival rate at $Q_i(j)$,

$\rho_i(j)$: traffic load at $Q_i(j)$, $\rho_i(j) = \bar{x} \cdot \lambda_i(j)$,

ρ_i : traffic load at node i , $\rho_i = \sum_{j=1}^N \rho_i(j)$,

$D_i(j)$: average queuing delay at $Q_i(j)$.

Since symmetric traffic pattern has been assumed, $\lambda_i(j)$ can be obtained as:

$$\lambda_i(j) = \begin{cases} \lambda(\frac{j-i+1}{N-1}), & \text{if } i < j \\ \lambda(\frac{N-1-i+j}{N-1}), & \text{if } i > j \\ \lambda, & \text{if } i = j \end{cases} \quad (4.15)$$

To obtain average waiting time $D_i(j)$, we use the approximation proposed in [58] as a starting point to derive the average end-to-end delay for VSQ.

$$D_i(j) = \frac{(1 - \rho_i(j))(1 - \rho_i) + \frac{\rho_i(j)}{w_i(j)}(2 - \rho_i)}{1 - \rho_i - \frac{\lambda_i(j)s}{w_i(j)}} \cdot \frac{T_i + \frac{s}{1 - \rho_i} \sum_{l=1}^N \frac{\rho_i^2(l)}{w_i(l)}}{\sum_{l=1}^N \left[\rho_i(l)(1 - \rho_i(l)) + \frac{\rho_i^2(l)(2 - \rho_i)}{w_i(l)(1 - \rho_i)} \right]}, \quad (4.16)$$

where

$$T_i = \frac{\rho_i \sum_{l=1}^N (\lambda_i(l) \cdot x^{(2)})}{2(1 - \rho_i)} + \frac{\rho_i s^{(2)}}{2s} + \frac{s}{2(1 - \rho_i)} [\rho_i^2 - \sum_{l=1}^N \rho_i^2(l)],$$

and

s : the first moment of the switchover time between queues,

$s^{(2)}$: the second moment of the switchover time between queues.

For simplicity, we assume the switchover time is negligible and hence, $s = s^{(2)} = 0$.

When node i sends a packet to node j , the following path is traversed:

$$i \rightarrow (i+1) \text{ MOD } N \rightarrow (i+2) \text{ MOD } N \rightarrow \dots \rightarrow j.$$

By definition, the average access delay at node i , $D_{VSQ}^{Access}(i)$, is the same as $D_i(i)$, i.e., the delay at the add-on queue. For a balanced network, all $D_{VSQ}^{Access}(i)$'s are equal, i.e., the packets from different nodes experience the same access delay, hence, the node index i could be dropped. Without loss of generality, we choose $i=1$ and the average access delay for VSQ ring, D_{VSQ}^{Access} , is given as:

$$D_{VSQ}^{Access} = D_1(1) \quad (4.17)$$

The average end-to-end queuing delay from source node i to destination node j (assume $i \leq j$), represented by $D_{i \rightarrow j}$, is the summation of the average queuing delay at all the nodes the packet traverses. This is because the queuing delays at each node are independent, and hence, additive.

$$\begin{aligned} D_{i \rightarrow j} &= \sum_{k=i}^j D_k(i) \\ &= D_i(i) + D_{(i+1)}(i) + \dots + D_j(i) \end{aligned} \quad (4.18)$$

For symmetric traffic pattern, a node sends its traffic to all other $(N-1)$ nodes with equal probability of $1/(N-1)$. Thus, the average end-to-end queuing delay for packets originating at node i , D_i , is obtained by averaging over all the $(N-1)$ possible destinations:

$$D_i = \frac{\sum_{j=1}^{N-1} D_{i \rightarrow (i+j) \bmod N}}{N-1} \quad (4.19)$$

For a balanced network in which $\lambda_i(i) = \lambda$ for $1 \leq i \leq N$, all D_i 's are equal, i.e., the packets from different nodes suffer the same end-to-end queuing delay. Hence, the node index i could be dropped. Without loss of generality, we choose $i=1$ and derive the average end-to-end queuing delay, D , as follows:

$$\begin{aligned}
D = D_1 &= \frac{\sum_{j=1}^{N-1} D_{(1 \rightarrow (1+j) \text{ MOD } N)}}{N-1} = \frac{\sum_{j=1}^{N-1} \sum_{k=1}^{(1+j) \text{ MOD } N} D_k(1)}{N-1} = \frac{\sum_{i=1}^1 D_i(1) + \dots + \sum_{i=1}^{N-1} D_i(1)}{N-1} \\
&= \frac{D_1(1) + (D_1(1) + D_2(1)) + \dots + (D_1(1) + D_2(1) + \dots + D_{N-1}(1))}{N-1} \\
&= \frac{1}{N-1} \sum_{i=1}^{N-1} (N-i) \cdot D_i(1),
\end{aligned} \tag{4.20}$$

where $D_i(1)$ can be obtained using (4.16).

The average end-to-end delay for VSQ ring, D_{VSQ}^{e2e} , consists of the queuing delay and transmission delay at every traversed node. The average end-to-end queuing delay, D , is derived in (4.20). As stated in Section 4.7.1, the propagation delay is neglected and the transmission time at every node is independent of each other, hence, the end-to-end packet delay is given by:

$$D_{VSQ}^{e2e} = D + \bar{x} \cdot H \tag{4.21}$$

where D is given in (4.20) and H is the average number of hops the packet traverses. In symmetric traffic pattern, each of the N nodes sends traffic to the other nodes with equal probability of $P_i = 1/(N-1)$, for $i = 1..N-1$. Thus, we have:

$$H = \sum_{i=1}^{N-1} i P_i = \frac{1}{N-1} \left(\frac{N-1}{2} (N-1+1) \right) = \frac{N}{2} \tag{4.22}$$

4.7.3 Average Delay Analysis for RPR

As described in Section 2.2.4, The RPR MAC is based on BIR. Every RPR node has an insertion buffer and a node buffer. The selection of the buffer from which a packet is transmitted depends on the priority of that buffer. RPR defines that the ring traffic has non-preemptive priority over the node traffic. Associating the ring traffic to class 1 and

node traffic to class 2, we can model the RPR node as the head-of-the-line non-preemptive queuing system [69][73][74] Let us first describe the general solution of the head-of-line non-preemptive queuing system and then apply the solution to RPR delay analysis.

For class k , we define:

$$W_k \triangleq E[\text{queuing time for class } k \text{ packet}] \quad (4.23)$$

$$D_k \triangleq E[\text{total time in system for class } k \text{ packet}] = W_k + \bar{x}_k \quad (4.24)$$

where $E[\cdot]$ is the expected value and $\bar{x}_k$ is the service (or transmission) time for class k packets.

The solution for the queuing time of a packet with priority k ($k = 1, 2, \dots, K$, with 1 being the highest priority and K being the lowest priority) is given by [69] as follows:

$$W_k = \frac{W_0}{(1 - \sum_{i=1}^{k-1} \rho_i)(1 - \sum_{i=k}^K \rho_i)} \quad (4.25)$$

where W_0 is the mean residual service time and ρ_i is the fraction of time that the server is busy serving packets from class- i . Since the arrival process is assumed to be Poisson process, ρ_i is equal to the probability that a customer finds a class- i customer is in service. Also, because of Poisson arrivals, the mean residual life of service time as observed by an arrival is equal to the second moment of service divided by two times the first moment [69]. The above statements allow us to calculate W_0 as:

$$W_0 = \sum_{i=1}^K \rho_i \frac{x_i^{(2)}}{2x_i} = \sum_{i=1}^K \frac{\lambda_i \cdot x_i^{(2)}}{2} \quad (4.26)$$

Substituting (4.26) into (4.25), we obtain:

$$W_k = \frac{\sum_{i=1}^K \frac{\lambda_i x^{(2)}}{2}}{(1 - \sum_{i=1}^{k-1} \rho_i)(1 - \sum_{i=1}^k \rho_i)} \quad (4.27)$$

Next, we apply (4.27) to an RPR node where the mean delay for the ring traffic is represented by W_1 , and the mean delay for the node traffic is denoted by W_2 respectively. The class index of i of $\bar{x}$ and $x^{(2)}$ is dropped from (4.27) because the packet sizes and service times are the same for both ring traffic and node traffic. Due to the assumption of symmetric traffic pattern, the values of W_1 and W_2 are the same for every node and given by:

$$W_1 = \frac{(\lambda_1 + \lambda_2)x^{(2)}}{(1 - \lambda_2\bar{x})(1 - (\lambda_1 + \lambda_2)\bar{x})} \quad (4.28)$$

$$W_2 = \frac{(\lambda_1 + \lambda_2)x^{(2)}}{(1 - \lambda_2\bar{x})} \quad (4.29)$$

where λ_1 is the arrival rate of the ring traffic and λ_2 is the arrival rate of the node traffic.

By definition, we have $\lambda_2 = \lambda$ and we can derive λ_1 as follows.

Let $\lambda_{i \rightarrow j}$ denotes the average arrival rate of packets at node i that are destined to node j . Without loss of generality, we focus on the performance at node N . The average rate of packets that pass through node N , Λ_N , is given by:

$$\Lambda_N = \sum_{i=2}^{N-1} \sum_{j=1}^{i-1} \lambda_{i \rightarrow j} \quad (4.30)$$

For symmetric traffic pattern, we have:

$$\lambda_{i \rightarrow j} = \begin{cases} \frac{\lambda}{N-1}, & i \neq j \\ 0, & i = j \end{cases} \quad (4.31)$$

Substituting (4.31) to (4.30), we have:

$$\begin{aligned} \Lambda_N &= \sum_{i=2}^{N-1} \sum_{j=1}^{i-1} \lambda_{i \rightarrow j} \\ &= \sum_{i=2}^{N-1} \frac{(i-1)\lambda}{N-1} = \frac{\lambda}{N-1} \sum_{i=2}^{N-1} i - \frac{\lambda}{N-1} (N-1-1) \\ &= \frac{\lambda}{N-1} \sum_{i=1}^{N-1} i - \frac{\lambda(N-1)}{N-1} = \frac{\lambda}{N-1} \frac{N(N-1)}{2} - \lambda \\ &= \left(\frac{N}{2} - 1\right)\lambda \end{aligned} \quad (4.32)$$

In summary, for symmetric traffic pattern, we have:

$$\begin{cases} \lambda_1 = \left(\frac{N}{2} - 1\right)\lambda, \\ \lambda_2 = \lambda. \end{cases} \quad (4.33)$$

By definition, W_2 in (4.29) is the delay at a node and represents the average access

delay for RPR ring, D_{RPR}^{Access} :

$$D_{RPR}^{Access} = W_2 = \frac{(\lambda_1 + \lambda_2)x^{(2)}}{(1 - \lambda_2\bar{x})} \quad (4.34)$$

The average end-to-end delay for RPR ring, D_{RPR}^{e2e} , consists of two parts: the waiting and transmission time at the source node and the waiting and transmission times at all intermediate nodes. It can be written as:

$$D_{RPR}^{e2e} = W_2 + \bar{x} + (W_1 + \bar{x}) \cdot H \quad (4.35)$$

where W_1 and W_2 are given in (4.28) and (4.29) respectively. H is the mean number of intermediate nodes a packet traverses and derived in (4.22) as $N/2$.

4.7.4 Comparison Using Poisson Model

Based on the above analysis, we obtain some numerical results on the average access delay and average end-to-end delay for both RPR and VSQ schemes. We consider a uni-directional OC-12 rate ring with $N=8$ nodes. Packet size is set to be 500 bytes. Weights are assumed to be equal for all nodes in VSQ scheme.

Figure 4-19: Access delay comparison of RPR and VSQ

Figure 4-19 and Figure 4-20 show the access delay and end-to-end delay as a function of load for both RPR and VSQ schemes. The ring bandwidth utilization (i.e., system load) is at 100% when the traffic load at each node is 25%. This is because for a ring with spatial reuse, packets traverse half of the ring on average, i.e., the average distance between source and destination is four nodes for an eight-node ring.

As we can see from Figure 4-19, VSQ offers less access delay compared to RPR irrespective of the system load. As for the end-to-end delay, it can be seen from Figure 4-20 that when the system load is below 80%, VSQ offers less end-to-end delay; whereas

for higher load, VSQ has the same end-to-end delay as RPR. The reason is that at high load, the queuing delay dominates the end-to-end delay, thus, diminishing the advantage of VSQ on the access delay.

To further validate our delay analysis, we obtain simulation results of the access delay and end-to-end delay with the same parameters used in the numerical results. The simulation results (drawn with symbols) closely track the numerical results (plotted in lines) as shown in Figure 4-19 and Figure 4-20.

Figure 4-20: End-to-end delay comparison of RPR and VSQ

4.7.5 Comparison Using Bursty Model

Since there is no analytical delay solution for bursty traffic models, we conduct simulations to study the average delay for both VSQ and RPR. The bursty traffic model we use is ON/OFF Pareto traffic model due to its simplicity.

Figure 4-21: ON/OFF Pareto traffic model

As illustrated in Figure 4-21, when the model is at the ON state, it generates traffic at a constant rate; when the source is at the OFF state, no packet is generated. By proper choice of its parameters, the ON/OFF model can represent certain levels of burstiness in the traffic it generated. We chose the sojourn time distribution of both ON and OFF states as heavy-tailed Pareto distribution with the probability density function and probability distribution function as:

$$f(x) = \frac{\alpha\beta^\alpha}{x^{\alpha+1}} \quad (4.36)$$

$$F(x) = 1 - \left(\frac{\beta}{x}\right)^\alpha \quad (4.37)$$

where $\alpha > 0$ and $\beta > 0$ represent the shape parameter and the scale parameter of the Pareto distribution, respectively. For $\alpha > 1$, the mean is defined as:

$$\bar{x} = \frac{\alpha\beta}{\alpha - 1}. \quad (4.38)$$

The relation between α and the Hurst parameter, H , which represents the burstiness of the traffic is given by [81]:

$$\alpha = 3 - 2H \quad (4.39)$$

Because the Hurst parameter ranges from 0.5 to 1, α ranges from 1 to 2. The variance does not converge in this region meaning that the traffic is bursty. We set α to 1.1 for the

sojourn time distribution of both ON and OFF states. This results in the average ON and OFF time of 11 msec using (4.38). From these numbers, we compute the peak to average ratio as 2. Other parameters used in the simulations are set as follows.

- Packet size = 500 Bytes
- Ring nodes = 10,
- link rate = 622 Mbps,
- Symmetric traffic pattern,
- Packet interarrival time is exponentially distributed. Different system load is set by changing the parameters of the distribution of the interarrival times.

Figure 4-22 Figure 4-23 show the access delay and end-to-end delay of RPR and VSQ with the Poisson model. As seen from the plots, VSQ features both less access delay and end-to-end delay compared to RPR when add-on traffic follows bursty ON/OFF model.

Figure 4-22: Access delay comparison using Pareto traffic model

Figure 4-23: E2E delay comparison using Pareto traffic model

4.8 Transit Buffer Requirement

4.8.1 Analysis

In both VSQ and RPR 2-TB scheme, the LP transit traffic is used to store LP transit traffic in order to reduce the access delay of add-on traffic. However, the main difference between these two schemes is that VSQ virtually allocates this LP buffer among all nodes through virtual source queues to ensure “source-based fairness”. While in RPR, the LP buffer is just a First-Come-First-Serve (FCFS) buffer.

One common requirement for VSQ and RPR is to ensure a loss-less transit path. However, VSQ and RPR employ different mechanisms to achieve this goal. Therefore, the required transit buffer size is different. As described in Section 3.3.4, VSQ employs “one-hop-backpressure” flow control while RPR depends on the fair rate feedback

control to avoid packet loss on the transit path. In this section, we study the LP transit buffer requirement since it directly relates to the node complexity.

- **VSQ Buffer Requirement**

Recall that there are two thresholds for virtual source queue k at node i : $VSQHighThreshold_k(i)$ and $VSQLowThreshold_k(i)$ defined in the OHB flow control scheme. For simplicity, we assume all virtual sources of all nodes use the same high and low thresholds as $VSQHighThreshold$ and $VSQLowThreshold$. Once the incoming traffic to a virtual source queue exceeds $VSQHighThreshold$, a one-hop-backpressure signal is sent to the upstream neighboring node to stop its input to this specified source-based queue. To achieve a lossless transit path, buffer engineering of a VSQ node is required to hold all the traffic before the one-hop backpressure signal reaches the upstream neighboring node. The time it takes for the signal reaches the upstream neighbor is *one-hop-delay* and the maximum traffic rate for this period is the unreserved rate, C . Therefore, the maximum transit buffer size that a VSQ node needs to ensure no packet loss, $sizeVSQ$, is given by:

$$sizeVSQ = VSQHighThreshold \cdot N + one_hop_delay \cdot C. \quad (4.40)$$

- **RPR Buffer Requirement**

The buffer size requirement for RPR has been studied in [15]. Once the $stqHighThreshold$ is crossed, no more LP local traffic is allowed to access the ring and hence, a feedback control message is sent to upstream nodes. The worst case scenario is when all the incoming transit traffic are LP with *lineRate* and the local node traffic are HP with *reservedRate* (R_{re}). Since HP node traffic has priority than the LP transit traffic,

the transit buffer size grows at the rate of R_{re} . Therefore, the maximum transit buffer size at an RPR node to ensure no packet loss, $SizeRPR$, can be obtained as:

$$SizeRPR = stqHighThreshold + R_{re} \cdot responseTime \quad (4.41)$$

The second term in (4.41) is the maximum LP transit traffic that the congested node is expected to receive during the $responseTime$ that has been given in Equation (4.2). Equation (4.41) can also be used to estimate the maximum amount of HP traffic that can be accommodated. For a fixed buffer size, the maximum HP traffic rate the system can accommodate, R_{re} , is given by:

$$R_{re} \leq (SizeRPR - stqHighThreshold) / responseTime \quad (4.42)$$

Let us compare the transit buffer size requirement of VSQ and RPR through the following example.

Example 4-1: Consider a ring with $N=100$ nodes, total ring length $D_r=1000km$, every hop distance of 10km, $line\ rate=622Mbps$, and $R_{re}=100Mbps$.

According to the threshold setting guidelines in section 3.6.2, we have:

$$VSQHighThreshold > VSQLowThreshold > one-hop-delay * (line\ rate - R_{re}) = 3.25KB.$$

The above requirement is set in order to achieve maximum utilization of the ring bandwidth. We set $VSQHighThreshold = 10KB$, which is three times larger than the above minimum requirement.

We plug the data into Equation (4.40) to obtain $sizeVSQ$ and its functional relationship with the number of nodes, N .

$$\begin{aligned} sizeVSQ &= 10KB * N + 5\mu s/km * 10\ km * 622Mbps/8 \\ &= 0.01N + 0.004\ (MB) \end{aligned}$$

$$=1.004 \text{ (MB)}.$$

For the sake of fair comparison, the RPR threshold $stqHighThreshold$ needs to be N times of that of each virtual queue of a VSQ node, hence,

$$stqHighThreshold = 10KB * N = 1MB.$$

First we estimate the $responseTime$ using Equation (4.2) and obtain the following:

$$\begin{aligned} responseTime &= agingInterval + maxFRTT + maxFRED \\ &= 100 \mu s + 10ms + 10KB * N * N / (linerate - R_{re}) \\ &= 1.51sec \end{aligned}$$

We then apply Equation (4.41) and some algebra to obtain $sizeRPR$ and its functional relationship with the number of nodes, N .

$$\begin{aligned} sizeRPR &= stqHighThreshold + R_{re} * responseTime \\ &= 1MB + 1.51sec * 100Mbps / 8 \\ &= 0.0019N^2 + 0.01N + 0.125 \text{ (MB)} \\ &= 20.125 \text{ (MB)}, \end{aligned}$$

which is about 20 times of the size of VSQ buffer requirement of 1.004 MB.

Figure 4-24 shows the buffer size requirement versus the number of ring nodes, N , for this example. It is noted that $sizeRPR$ grows with N^2 while $sizeVSQ$ grows linearly with N . The reason is that the feedback delay in RPR depends on the number of ring nodes while in VSQ, the feedback delay is “one-hop-delay” which does not depend on the number of nodes in the congestion span.

Figure 4-24: Transit buffer size requirement comparison for Example 4-1

4.8.2 Simulation Results on the Transit Buffer Occupancy

In this subsection, we compare the transit buffer occupancy of RPR and VSQ through simulations (scenario 9) to show that VSQ has less buffer occupancy due to the fast feedback control.

Simulation scenario setup:

- Packet size = 500 bytes,
- N=10 nodes,
- link rate = 622 Mbps.
- Traffic pattern: As shown in Figure 4-25, node 6 and node 7 and node 9 send traffic to HUB node 0. Node 6, 7 starts sending fairness eligible traffic at 0 seconds with rate of 311 Mbps. At 0.2 seconds, node 9 starts to add HP traffic of 100Mbps. We observe the transit buffer occupancy at the node 9 since the output link of node 9 is the bottleneck.

- Thresholds parameters:

$$VSQHighThreshold = 250KB,$$

$$VSQLowThreshold = 200 KB,$$

$$stqHighThreshold = VSQHighThreshold * 2 = 500KB \text{ (equivalent to the 250KB}$$

per source queue in VSQ since there are two source queues in this scenario),

$$stqLowThreshold = 400 KB.$$

Figure 4-25: Simulation setup to study buffer occupancy

Simulation results and analysis:

In Figure 4-26, we show the evolution of buffer occupancy at the bottleneck point, the output link of node 9, link 9-0. At the instant when node 9 starts to add traffic, additional buffers are used up at the bottleneck due to the delay in adjusting the input rate to the reduced service rate. Because the smaller delay in the feedback cycle time, VSQ transit buffer is less occupied compared to RPR. The VSQ transit buffer occupancy is nearly bounded around the buffer threshold of 500 KB, while the RPR transit buffer occupancy fluctuates much more, up to 625 KB, due to the larger feedback delay.

Figure 4-26. Transit buffer occupancy comparison: VSQ vs RPR

We conduct a simple analysis on the simulation results. With VSQ, the OHB scheme is activated at node 9 once the virtual queue 6 and 7 reaches *VSQHighThreshold*. A notification is sent to node 8 to block the service of the virtual queues 6 and 7. Once the notification is received by node 8, the transit buffer of node 9 stops growing. The “one-hop-delay” is $50\mu s$ for a $10km$ link. The traffic arrival rate to virtual source queue 6 and 7 at node 9 is 311 Mbps, and the service rate is $(622-100)/2=261Mbps$. As a result, the transit buffer size grows at the rate of $311-261=50$ Mbps. We can estimate the transit buffer size as follows.

$$(250 KB + (50Mbps * 50\mu s) / 8) \cdot 2 \cong 501 KB$$

Once the backpressure message takes effect after the one-hop-delay, the buffer at the node 9 starts to clear. By the time the size of virtual queues at node 9 falls below the

VSQLowThreshold, the service rate at node 8 resumes to 311 Mbps. Therefore, the buffer occupancy at node 9 is nearly upper bounded by 501 KB.

On the other hand, in RPR, the HP node traffic of node 9 has priority over the LP transit traffic from nodes 6 and 7. The LP transit buffer grows once node 9 starts adding HP traffic to the ring. When the *stqHighThreshold* is reached, a feedback message containing the fair rate is sent to all the upstream nodes, i.e., nodes 6 and 7. The transit buffer keeps growing until the feedback message takes into effect, i.e., the fair rate has been received and applied by node 6 and node 7. Since these nodes are a few hops away from node 9, RPR feedback messages encounter more delay than OHB feedback messages, therefore, the transit buffer occupancy level of RPR is much higher than in VSQ.

It is clear from Figure 4-26 that the OHB scheme is more successful at keeping the sum of the buffer occupancy at or around the desired levels than the RPR feedback control scheme. This happens because the end-to-end delay with the RPR scheme is much larger than the feedback delay with OHB.

The above simple example is for illustration purposes only. With larger ring size, the difference in buffer occupancy between RPR and VSQ becomes even larger. This is because larger ring size means larger propagation delay and large *responseTime*, which results in larger transit buffer occupancy. On the other hand, the transit buffer occupancy of VSQ only depends on the one-hop-delay not the size of the ring. This is illustrated in the following subsection.

4.8.3 Simulation Results on the Transit Buffer Occupancy Variance

The simulation setup in Scenario 10 is similar to the Scenario 9 (Figure 4-24) in the above section except that there is only one source that contributes to the congestion at node 9 and the source is set to be one to five hops away from node 9.

In Figure 4-27, we plot both the transit buffer occupancy average and variance versus the number of hops between the source and the congested node 9 (bottleneck). We see that when the number of hops is one, RPR has the same values as VSQ does. This is because the feedback delay is the same for VSQ and RPR when the number of hops is one. However, when the number of hops increases, the transit buffer occupancy average and variance of RPR grows but that of VSQ does not grow [78].

Figure 4-27: Transit buffer average and variance comparison: VSQ vs. RPR

4.9 Summary

In this chapter, we first outlined both the performance and the MAC design compliancy metrics. We then justified our performance evaluation methodology and presented detailed delay analysis and simulation results from a suite of ten benchmark scenarios. Here is a summary of our concluding remarks.

Our overall results are encouraging. We have shown that VSQ is not only compliant with the desired features of MAC for packet rings but also it outperforms RPR in all the comparison scenarios. Further, the OHB control scheme consistently outperforms the competing RPR feedback control in terms of transit buffer requirement and responsiveness. Thus, we claim that our venture to design robust and efficient MAC scheme for packet rings has been successful. We justify this claim by reviewing the results from the analysis and simulations.

- The results of scenarios 1 to 4 (in Table 4-3) showed that VSQ ensures weighted fairness, achieves spatial reuse, allows bandwidth sharing through fast reclaiming of the unused bandwidth and provides isolation between well-behaving sources and ill-behaving ones. The results validate our early claim that VSQ guarantees fairness while maximizing link utilization.
- The results of scenario 5 showed that under an unbalanced traffic pattern, VSQ has no bandwidth oscillation and, hence, it does not suffer from throughput loss. This is in contrast with what RPR offers under the same traffic condition.
- Scenario 6 examined the convergence time to the fair rate. We found that the fair rate convergence time of VSQ is much less than that of RPR and its enhancement of DVSR. Moreover, the fair rate convergence time in RPR and DVSR depends

on the RTT while for VSQ, the RTT has no impact on the fair rate convergence time.

- Scenario 7 validated our delay analysis based on the assumptions of Poisson traffic arrivals and uniform traffic pattern. It shows that the simulation results closely track the numerical results derived for the average access delay and average end-to-end delay for both VSQ and RPR schemes. The close match of the two data sets indicates that both the simulation program and analysis work properly.
- Scenario 8 used simulation method to assess the performance of VSQ and RPR with bursty traffic. This is because there is no analytical solution for these traffic models. We use Pareto ON/OFF bursty traffic model to compare the average access delay and average end-to-end delay of both VSQ and RPR schemes. The results show that VSQ offers less average access delay than RPR. In terms of average end-to-end delay, it consistently either matches or outperforms RPR.
- Scenario 9 studied the transit buffer requirement of both VSQ and RPR. It shows a VSQ node requires less transit buffer than an RPR one to ensure lossless transit path. This is due to the merits of the OHB such as less delay in the control loop.
- Scenario 10 compared the buffer occupancy variance between VSQ and RPR. The results show that when the distance between a congested node and the source node increases, both the average and variance of the transit buffer occupancy grows in RPR but does not grow in VSQ.

To summarize, we have shown that VSQ works well as a MAC scheme for packet rings. This conclusion is valid to the extent that our suite of benchmarks is

comprehensive. While we have tested for ill-behaving users, severe buffer contention, Poisson traffic and bursty traffic models and multiple bottlenecks, we may have ignored some other (perhaps equally important) factors such as: two-way traffic. Thus, these limitations must be borne in mind while reviewing our conclusions. We recognize that no suite of scenarios, at least at the current state of the art, can claim to be comprehensive. We have tried our best to create worst-case scenarios that test specific problems in fairness control schemes. It is possible that some of the factors we have ignored are critical in determining protocol performance, but this is still a matter for speculation. Developing a more comprehensive suite of benchmarks can be a subject for future research.

To conclude, while our simulations are only for a small suite of scenarios, and each scenario only has a small number of nodes, we feel that VSQ has several promising features that can make it suitable for metro packet ring networks. Studying its effectiveness in the real world is an area for future work.

CHAPTER 5 Conclusions and Recommendations for Future Research

5.1 *Thesis Summary*

In Chapter 1, we addressed the design goals and the challenges of MAC schemes for packet ring networks. We then reviewed the existing MAC schemes including the IEEE RPR standard and its enhancements in Chapter 2. In reference to the MAC design goals, the RPR MAC and its enhancements suffer certain performance deficiencies which include throughput loss and large access delay. This motivates our research goal: design of a MAC scheme that realizes all the design goals including fairness and maximum throughput. In Chapter 3, we pursued this research goal by proposing VSQ scheme that provides guaranteed fairness, maximum throughput and low access delay. We also proposed OHB as an efficient flow control scheme to ensure lossless transit paths. OHB exploits the source information provided by the VSQ and effectively control the transit buffer occupancy by implicitly controlling the input rate of the traffic sources. In Chapter 4, we considered various scenarios to compare the performance of VSQ with that of the RPR standard. Our results show that VSQ not only has met all the MAC design goals set in Chapter 1, but also has the same or better performance compared to the RPR MAC.

5.2 MAC Design Goals Re-examined

We now re-examine the seven design goals for a MAC scheme presented in Chapter 1, and use the analysis and simulation results of Chapter 4 to examine the extent that VSQ scheme has achieved in reference to these goals.

- **Fairness:** The VSQ scheme ensures fairness through a fair scheduler that allocates bandwidth fairly to all sources that share a link. Simulation results in Section 4.3 demonstrated this property. Thus, we claim that VSQ is a fair MAC scheme.
- **Efficiency:** We desire a MAC scheme to be efficient in two ways. First, its control overhead should not consume excessive amount of resources, such as network bandwidth and processing time. Second, it should maximize the utilization of network capacity.

VSQ requires little overhead on the net amount of data transferred in the network. This is because the OHB feedback messages are only nine bits long and are only sent upon threshold crossing with no calculation required. Whereas in RPR, every node has to measure the incoming rates, estimate the fair rate, and send the feedback control messages, which contain fair rate information and are sixteen bytes long, to all the upstream nodes in every control interval.

As illustrated in Chapter 3, RPR underutilizes the network bandwidth. VSQ, in contrast, efficiently utilizes the network bandwidth by using a fair scheduler that is work conserving, i.e., it does not keep bandwidth idle if there are packets waiting to be served. The simulation results in Section 4.4 have shown that VSQ

maximizes the throughput in all the scenarios. As a result, VSQ meets the two efficiency criteria that were set in Chapter 1.

- **Fast fairness convergence:** Since VSQ does not require fair rate estimation and feedback to ensure fairness, the convergence to fair rate is faster than the RPR. In addition, the convergence time of VSQ is independent of the round trip time of the congestion domain. The achievement of this goal has been demonstrated in Section 4.5.
- **Low access delay for node traffic:** Because VSQ provides guaranteed fair share to each node, the local add-on traffic can get on the ring with minimal delay. While with RPR, the local traffic can only be added to the ring after the feedback control messages have taken into effect. This has been shown through analysis and simulations in Sections 4.6 and 4.7.
- **Scalability:** Scaling is required in two dimensions: bandwidth and ring size in terms of number of nodes. First, higher ring bandwidth translates to larger bandwidth-delay product, therefore, sources can no longer ignore the propagation delay in conducting a flow control. However, the VSQ feedback control does not depend on the round trip delay between source and destination but only depend on the one-hop delay. As a result, unlike RPR, an increase in ring bandwidth does not impact VSQ performance as its operation is almost independent of the bandwidth-delay product. Second, as the number of ring nodes increases, RPR suffers longer access delay and more throughput loss because RPR feedback control depends on the round trip delay between source and destination. Whereas the impact on VSQ is just the implementation of the same number of virtual

source queues. The RPR standard limits the maximum number of nodes to 256. Thus, the number of virtual source queues in a node is limited to 256. As a result, we claim that the VSQ MAC scheme scales well with both bandwidth and ring size.

- **Ability to deal with ill-behaving sources:** Section 4.3.4 adequately answers this requirement by showing the isolation property of the VSQ. As we see from the simulation results, unlike RPR, there is no impact from ill-behaving sources on the well-behaving ones in the VSQ scheme. This is because VSQ isolates traffic from different sources. Upon congestion, only the ill-behaving nodes suffer performance degradation, not the well-behaving ones.
- **Simplicity:** The simplicity requirement means that a MAC scheme is easy to specify and implement. Both VSQ and OHP are conceptually simple, and no signaling or rate control mechanism is needed to ensure fairness.

VSQ implementation may look complex at first glance, and it seems as if it may add delay and buffer space. However, as described in Chapter 3, the complexity added by the virtual source queues in VSQ can be offset by the elimination of the VDQs that are required in the RPR standard to avoid HOL blocking. In addition, as we showed in Chapter 4, the delay of VSQ is not more than RPR; the buffer occupancy is actually reduced with VSQ, thanks to the efficiency of OHB that allows better utilization of the distributed buffers across the ring. VSQ requires per-flow queuing, but, as described in Chapter 3, this can be implemented by a single ordered linked list. There is nothing specific to VSQ that makes it any harder to implement than any other equivalent protocol and thus,

we feel confident that the benefits of having virtual source queuing outweigh its costs.

Based on the above analysis, we conclude that the VSQ scheme satisfies the MAC design goals listed in Chapter 1. While the same conclusion could not be drawn with the existing MAC schemes proposed in the standard and literatures. Table 5-1 summarizes the performance comparison between different MAC schemes.

Table 5-1: Relative performance comparison of different MAC schemes

Yes. HP traffic is supported through bandwidth reservation; LP traffic is supported through fair sharing	Same as RPR	Same as RPR	Same as RPR
Yes	Yes	Yes	No
Yes	Yes	Yes	Yes
Yes	Yes	Yes	Yes
No	No	No	Yes
Long	Medium	Medium	Short
Large	Large	Large	Small
Yes	Yes	Yes	No
Large	Large	Large	Small
Large	Large	Large	Small

5.3 VSQ Clarifications

Our performance results present a convincing case and demonstrate that VSQ is an efficient MAC scheme for metro packet ring networks. We clarify some potential misconceptions about VSQ in the following.

- *VSQ does not increase transit buffer*

As demonstrated in Section 4.8, VSQ MAC enables OHB flow control that results in less transit buffer occupancy and variance than those of RPR MAC. This is because the feedback cycle time of OHB is smaller, and hence, fewer packets are accumulated at the bottleneck buffer compared to the RPR MAC.

- *VSQ does not increase delay*

Our delay analysis and simulations in Chapter 4 show that this is not the case. On the contrary, since VSQ MAC ensures fair access and maximum throughput through fair scheduler, its access delay is significantly reduced and its end-to-end delay is either the same or reduced when compared to the RPR MAC.

- *VSQ does not increase complexity*

At first glance, this solution seems to imply drastic changes in the ring architecture. This is not so if we take a closer look as the following. The additional cost of the VSQ (fair scheduler and queue management) is marginal, particularly considering that RPR 2-TB nodes already support large LP transit buffer, and hence only software upgrade is needed towards VSQ. Efficient fair scheduler implementation is also widely available nowadays. Furthermore, the virtual source queues in the transit path eliminate the need for VDQ, which is

required at the ingress point by the RPR nodes to avoid HOL blocking (Section 3.4.3). The cost is further mitigated since OHB scheme requires fewer and shorter feedback control messages. This is because these control packets do not serve for fairness functions in contrast to the role they play in the RPR. Consequently, the processing overhead related to VSQ is much less. Table 5-2 summarizes the complexity comparison between different MAC schemes.

Table 5-2: Complexity comparison between different MAC schemes

RPR-AM: the add rate of the congested node; RPR-CM: the available bandwidth divided by the # of nodes.	Max-min approximation	Max-min approximation	Not needed
Yes (Use token bucket to enforce the fair rate received)	Yes (Use token bucket to enforce the fair rate received)	Yes (Use token bucket to enforce the fair rate received)	Not needed. (Instead, use implicit rate control through buffer management)
Explicit rate control	Explicit rate control	Explicit rate control	Implicit rate control
20	20	20	2
# of possible sources on the ring (max: 256)	# of possible sources on the ring (max: 256)	# of possible sources on the ring (max: 256)	0
2	2	2	# of possible sources on the ring (max:256)
No	Yes	Yes	No
Strict priority	Strict priority	Strict priority	Fair queuing

5.4 Contributions

This thesis has made a number of contributions to the research area of MAC for packet rings. We review some of them in this section.

The main contribution is designing the VSQ framework and the OHB flow control scheme. We believe that both schemes, though conceptually simple, have several advantageous features that make them suitable for packet ring networks. The VSQ scheme is shown to meet the MAC design goals for packet rings: it is efficient, fair, convergent and adaptive to changes in network configuration. It can provide better medium access control than other standard scheme or those proposed in the literatures.

Other contributions include developing analytical models of the packet ring networks and deriving the average access delay and end-to-end delay for both VSQ and RPR. Though the analysis is naive, the studies bring us insights into the delay comparisons of both MAC schemes. Cyclic model have been studied earlier for single node, our contribution lies in carrying out such analysis for ring networks with multiple nodes.

Furthermore, we have implemented the proposed scheme in a realistic network simulator, and conducted extensive simulation runs to study its behavior in various benchmark scenarios and compared the performance with that of the RPR MAC.

5.5 Recommendations for Future Research

While we believe that our work has several advantages, there are some weaknesses as well. Our theoretical models, though adequate for our comparison purposes, are rather naive. We believe that even better results can be achieved by using more sophisticated models for the analysis. Thus, this thesis is only a small step in providing practical solutions and analysis to fair medium access control in packet rings. Second, the simulations in Chapter 4, though extensive, are still far from a complete study of protocol behavior. While our choice of benchmark scenarios tries to test for several aspects of

MAC schemes, there are surely other aspects that we have overlooked. This is an area where additional work is required.

We suggest the following future research work:

- Studying transit buffer management: Since all flows share one physical buffer, dynamic allocation of the buffer seems to be an interesting problem.
- Investigating the mapping of SLA parameters (e.g., delay, throughput) to transit buffer threshold setting.
- Comparing the performance of various TCP protocols over VSQ and RPR.
- Inter-working with RPR nodes: We believe our scheme can inter-work with RPR nodes as they only differ in the scheduler and flow control. However, experiments need to be conducted to validate this claim.

Bibliography

- [1] T. S. Afferton, R. D. Doverspike, C. R. Kalmanek and K. K. Ramakrishnan, Packet-Aware Transport for Metro Networks, *IEEE Communications Magazine*, March 2004, pp.120-127.
- [2] Walter J. Goralski, SONET, Second Edition, *McGrawHill*, 2001.
- [3] GR-1230-CORE, SONET bi-directional line-switched ring equipment generic criteria, *Bellcore standard*, Issue 4, Dec. 1998.
- [4] Generic Framing Procedure (GFP) and Data over SONET and OTN, *American National Standard Institute*; 2002.
- [5] Virtual Concatenation of SONET paths, *ITU G707*, 2000.
- [6] Link Capacity Adjustment Scheme (LCAS): *ITU-T G. 7042/Y.1305*, June 2002.
- [7] Siwaruk Siwamogsatham, 10 Gigabit Ethernet Technology Overview, available at: <http://www.cse.wustl.edu/~jain/cis788-99/10gbe/index.html>.
- [8] Metro Ethernet Networks - A Technical Overview, available at Metro Ethernet Forum website: <http://www.metroethernetforum.org/PDFs/WhitePapers/metro-ethernet-networks.pdf>.
- [9] Rajiv Ramaswami, Kumar N. Sivarajan, Optical Networks: A Practical Perspective, *Morgan Kaufmann Publisher*, 2002.
- [10] Rapid Reconfiguration of Spanning Tree, *IEEE Standard 802.1w*, 2002.
- [11] I. Cidon, Y. Ofek, MetaRing - A Full-Duplex Ring with Fairness and Spatial Reuse, *IEEE Trans on Communications*, Vol. 41, No. 1, January 1993.
- [12] J. Chen, I. Cidon and Y. Ofek, A local fairness Algorithm for gigabit LAN's/MAN's with spatial reuse, *IEEE JSAC*, Vol. 11, No. 8, pages: 1183-1192, Oct. 1993.
- [13] An introduction to Resilient Packet Ring Technology, *a white paper by the Resilient Packet Ring Alliance*, Oct. 2001.
- [14] S. Spadaro, et. al.; Positioning of the RPR standard in contemporary operator environments, *IEEE Network*, , Volume: 18 , Issue: 2 , Mar-Apr 2004 Pages:35 – 40.
- [15] IEEE Standard 802.17: Resilient Packet Ring, June 2004, <http://ieee802.org/17>.

-
- [16] IEEE standard for token ring, *IEEE Standard* 802.5–1989.
 - [17] D. Picker, R.D. Fellman: Enhancing SCI's fairness protocol for increased throughput, *IEEE Int. Conf. On Network Protocols*, October, 1993.
 - [18] IEEE standard for a Scalable Coherent Interface (SCI), *IEEE Standard* 1596–1990.
 - [19] R.M. Needham, A.J. Herbert: The Cambridge Distributed Computing System. *Addison-Wesley*, London, 1982.
 - [20] I. F.E. Ross, Overview of FDDI: The Fiber Distributed Data Interface, *IEEE J. on Selected Areas in Communications*, Vol. 7, No. 7, September 1989.
 - [21] Specification of the ATMR Protocol (V. 2.0), *ISO/IECJTC1SC6 N7873*, January 1993.
 - [22] W.W. Lemppenau, H.R.van As, H.R.Schindler, Prototyping a 2.4 Gbit/s CRMA-II Dual-Ring ATM LAN and MAN, *Proc. of the 6th IEEE Workshop on Local and Metropolitan Area Networks*, 1993.
 - [23] A. Thomasian and H. Kanakia, Performance Study of Loop Networks Using Buffer Insertion, *Computer Networks* 3 (1979), pp. 419-425.
 - [24] W. Bux, M. Schlater, An Approximation Method for the Performance Analysis of Buffer Insertion Rings, *IEEE Trans. on Comm.*, vol. 31, no. 1, pp. 50-55, January, 1983.
 - [25] D. Bertsekas and R. Gallager, Data Networks, *Prentice Hall*, 1992.
 - [26] R. Simha and Y. Ofek, A starvation-free access protocol for a full-duplex buffer insertion ring local area network, *Computer Networks and ISDN Systems* 21 (1991) 109-120.
 - [27] V. Gambiroza, P. Yuan, L. Balzano, Y. Liu, S. Sheafor and E. Knightly, Design, Analysis and Implementation of DVSR: A Fair, High Performance Protocol for Packet Rings, *IEEE/ACM Trans. on Networking*, vol. 12, no. 1, Feb., 2004.
 - [28] P. Yuan, V. Gambiroza, E. Knightly, The IEEE 802.17 Media Access Protocol for High-speed Metropolitan-area Resilient Packet Rings, *IEEE Network*, Volume 18, Issue 3, May-June 2004, Pages:8 – 15.
 - [29] F. Davik, M. Yilmaz, S. Gjessing, and N. Uzun, IEEE 802.17 Resilient Packet Ring Tutorial, *IEEE Communications Magazine*, Volume: 42 , Issue: 3 , Mar 2004, Pages:112 – 118.
 - [30] IEEE standard for Wireless LAN, *ANSI/IEEE. ANSI/IEEE Std. 802.11, 2000 Edition*, 2000.

-
- [31] ANSI/IEEE. ANSI/IEEE Std. 802.3, 2000 Edition, 2000. *IEEE standard for Ethernet*.
 - [32] Information technology - Open System Interconnection, *ISO/IEC 7498-1* 1994.
 - [33] P. McKenney, Stochastic Fairness Queuing, *Proceedings of IEEE INFOCOM '90*, vol.2, pp. 733 -740, 3-7 June 1990.
 - [34] Y. Robichaud and C. Huang, Improved Fairness Algorithm to Prevent Tail Node Induced Oscillations in RPR, *Proc. IEEE ICC 2005*, Soeul, Korea, May 2005.
 - [35] D. Wang, K.K. Ramakrishnan, and C. Kalmanek, Congestion Control in Resilient Packet Rings, *Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP)* 2004., 5-8 Oct. 2004, Page(s):108 – 117.
 - [36] F. Davik, and S. Gjessing, The stability of the Resilient Packet Ring Aggressive Fairness Algorithm, *Proc. of the 13th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN)*, 2004, 25-28 April 2004, Page(s):17 – 22.
 - [37] M. J. Francisco, F. Yuan, C. Huang, and H. Peng, A Comparison of Two Buffer Insertion Ring Architectures with Fairness Algorithms, in *Proceedings of ICC'03*, Anchorage, May, 2003.
 - [38] K. Fendick, M. Rodrigues and A. Weiss, Analysis of a Rate-based Control Strategy with Delayed Feedback, *Proc. ACM SIGCOMM 92*, Aug. 1992, pp. 136--148.
 - [39] A. Shokrani, S. Khorsandi, I. Lambadaris, and L. Khan, Virtual Queuing, An Efficient Algorithm for Bandwidth Management in Resilient Packet Rings, *Proc. IEEE ICC 2005*, Soeul, Korea, May 2005.
 - [40] F. Alharbi and N. Ansari, A Novel Fairness Algorithm for Resilient Packet Ring Networks with Low Computational and Hardware Complexity, *Proc. of the 13th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN)*, 2004., 25-28 April 2004, Pages:11 – 16.
 - [41] F. Alharbi and N. Ansari, Distributed Bandwidth Allocation for Resilient Packet Ring Networks, *Computer Networks*, 49 (2005) pp. 161-171.
 - [42] A. K. Parekh, and R. G. Gallager, A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: the Multiple Node Case, *IEEE/ACM Trans. on Networking*, vol. 2, no. 2, pp. 137-150, April 1994.
 - [43] S. Keshav, An Engineering Approach to Computer Networking, *Addison Wesley*, 1997.
 - [44] A. Charny, D. Clark and R. Jain, Congestion Control with Explicit Rate Indication, *Proc. of IEEE ICC 1995*, pp. 1954—1963.

-
- [45] H. Tang and I. Lambadaris, Novel Bandwidth Allocation Scheme for Ring Networks with Spatial Reuse, *Proc. of the SPIE International Symposium: Asia-Pacific Optical and Wireless Communications (APOC)*, Vol. 4908, pp. 162-169, Shanghai, China, October 14-18, 2002.
 - [46] H. Tang and I. Lambadaris, Performance Evaluation of a New Fairness Control Scheme for Ring Networks with Spatial Reuse, *Proc. of the SPIE International Symposium: Information Technologies and Communications (ITCom)*, Vol. 5244, pp. 143-153, Orlando, USA, 7-11 September 2003.
 - [47] H. Tang, I. Lambadaris, H. Mehrvar, J. Talim and A. Matrawy, A New Access Control Scheme for Metropolitan Packet Ring Networks, *Proc. of IEEE Globecom 2004 Workshop (CAMAD '04)*, pp. 281 – 287, Dallas, Texas, USA, November 2004.
 - [48] W. Wong, H. Tang, and V. Leung, “Token Bank Fair Queuing: A New Scheduling Algorithm for Wireless Multimedia Services”, *International Journal of Communication Systems*, vol. 17, pp. 591-614, Aug. 2004.
 - [49] B. Bensaou, D. Tsang and K. T. Chan, Credit-based Fair Queuing (CBFQ): a Simple Service-scheduling Algorithm for Packet-switched Networks, *IEEE/ACM Trans. on Networking*, vol. 9, no. 5, pp. 591-604, October, 2001.
 - [50] M. Shreedhar and G. Varghese, Efficient Fair Queuing using Deficit Round Robin, *IEEE/ACM Trans. On Networking*, vol. 4, no. 3, pp. 375–385, Jun. 1996.
 - [51] J. Wang and Y. Levy, Managing Performance using Weighted Round Robin, *Proceedings of the fifth IEEE Symposium on Computers and Communications*, Page(s): 785 –792, 2000.
 - [52] J. Nagle, On Packet Switches with Infinite Storage, *IEEE Transactions on Communications*, vol. 35, no. 4, April 1987.
 - [53] S. J. Golestani, A Self-clocked Fair Queuing Scheme for Broadband Application, in *Proc. IEEE INFOCOM*, Toronto, Canada, Jun. 1994, pp. 636–646.
 - [54] J. Liang, T. N. Arvanitis, S. Woolley, Fair Weighted Round Robin Scheduling Scheme for DiffServ Networks, *IEEE Electronics Letters*, Volume: 39 , Issue: 3, Feb. 2003, page(s): 333 – 335.
 - [55] J Zhu, A. Matrawy I. Lambadaris and M. Ashourian, Optimization of Resilient Packet Ring Networks Scheduling for MPEG-4 Video Streaming, *Proc. IEEE ICC 2005* Soeul, Korea, May 2005.
 - [56] K. Ramakrishna and R. Jain, A Binary Feedback Scheme for Congestion Avoidance, *ACM Trans. on Computer System*, Vol. 8, No. 2, May 1990.

-
- [57] E. L. Hahne and R. G. Ballager, Round Robin Scheduling for Fair Flow Control in Data Communication Networks, *Proc. of IEEE ICC*, June 1986, pp103-107.
 - [58] M. Gerla and L. Kleinrock, Flow Control, a Comparative Survey, *IEEE Trans. on communications*, Vol. Com-28, No. 4, April 1980.
 - [59] L. Pouzin, Methods, Tools, and Observations on Flow control in Packet-Switched Data Networks, *IEE Trans. on comm.*, Vol. COM-29, No. 4, April 1981.
 - [60] M. Reiser, Queueing and Delay Analysis of a Buffer Pool with Resume Level, *Proceedings of the 9th International Symposium on Computer Performance Modelling, Measurement, and Evaluation* (College Park, Md., May 1983). A. K. Agrawala and S. K. Tripathi, Eds. *North-Holland*, New York, 1983, pp. 25-32.
 - [61] T. P. Yum, and H. M. Yen, Design Algorithm for a Hysteresis Buffer Congestion Control Strategy, *Proc. of the IEEE ICC* (June 1983). New York, pp. 499-503.
 - [62] S. Chen, M. Shyu, C. Zhan, and S. Peeta, A Novel Rate-based Hop by hop congestion control algorithm, *Proc. IEEE International Conference on Multimedia and Expo (ICME)*, 27-30 June 2004, Taipei, Taiwan.
 - [63] M. Schroeder, Andrew D. Birrell, Michael Burrows, Hal Murray, Roger M. Needham, Thomas L. Rodeheffer, Edwin H. Satterthwaite, and Charles P. Thacker, Autonet: A High-speed, Self-configuring Local Area Network Using Point-to-point Links, *IEEE JSAC*, Vol. 9, No. 8, Oct. 1991, pp 1318-1335.
 - [64] P. Mishra and H. Kanakia, A Hop by Hop Rate-based Congestion Control Scheme, *Proc. of ACM SIGCOMM*, pp. 112-123, Aug. 1992.
 - [65] J. Nagle, Congestion Control in TCP/IP Internetworks, *Computer Comm. Rev.* 14, 4 (Oct. 1984), 11-17.
 - [66] A. Choudhury and E. Hahne, Dynamic Queue Length Thresholds in a Shared Memory ATM Switch, *Proc. of IEEE INFOCOM*, pp. 679--687, April 1996.
 - [67] Y. Tamir and G. L. Frazier, Dynamically Allocated Multi-Queue Buffers for VLSI Communication Switches, *IEEE Trans. Computers*, vol. 41, no. 6, pp.725-737, June 1992.
 - [68] E.L. Hahne and A.K. Choudhury, Dynamic Queue Length Thresholds for Multiple Loss Priorities, *IEEE/ACM Transactions on Networking*, Volume 10, Issue 3, June 2002 Page(s):368 – 380.
 - [69] L. Kleinrock, Queueing Systems, Vol. 2: Computer Applications, *Wiley-Interscience*, 1976.

-
- [70] K. K. Leung, Cyclic-service Systems with Probabilistically Limited Services, *IEEE JSAC*, vol. 9, pp. 185-193, 1991.
 - [71] O. J. Boxma and W. P. Groenendijk, Pseudo-conservation Laws in Cyclic Service Systems, *J. App. Prob.*, vol. 24, pp. 949-964, 1987.
 - [72] S. F. Fuhrmann and Y. T. Wang, Analysis of Cyclic Service Systems with Limited Service: Bounds and Approximations, *Performance Evaluation*, no. 9, pp. 35-54, 1988.
 - [73] D.A. Schupke, Packet Transfer Delay of the SRP Ring, *IEEE Conference on Local Computer Networks (LCN)*, Tampa, Florida, pp. 464-465, November 15-16, 2001.
 - [74] D.A. Schupke, A. Riedl, Packet Transfer Delay Comparison of a Store-and-Forward and a Cut-Through Resilient Packet Ring, *International Zurich Seminar on Broadband Communications (IZS)*, Zurich, Switzerland, February 19-21, 2002.
 - [75] OPNET Modeler brochure, available at:
<http://www.opnet.com/products/brochures/Modeler.pdf>.
 - [76] D. Ferrari, Computer Systems Performance Evaluation, *Prentice Hall*, Englewood Cliffs, New Jersey, 1978.
 - [77] J. Banks, J. Carson and B. Nelson, Discrete-event System Evaluation, *Prentice Hall*, Upper Saddle River, New Jersey, 1999.
 - [78] H. Tang, I. Lambadaris and H. Mehrvar, Efficient Medium Access Control Using Virtual Source Queuing for Packet Rings, to be submitted.
 - [79] C. Huang, H. Peng and F. Yuan, A Deterministic Bound for the Access Delay of Resilient Packet Rings, *IEEE Communications Letters*, Volume 9, Issue 1, Jan. 2005, Page(s):87 – 89.
 - [80] S. C. Borst, O. J. Boxma and H. Levy, the Use of Service Limits for Efficient Operation of Multistation Single-medium Communication Systems, *IEEE/ACM Trans. on Networking*, vol. 3, no. 5, pp. 602-612, October 1995.
 - [81] W. Willinger, M. S. Taqqu, R. Sherman and D. V. Wilson, Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level, *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 71-86, February 1997.