Performance Analysis of Resilient Packet Rings with Single Transit Buffer

submitted by

Fengjie Yuan, B. Eng., M. Eng.

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

Master of Applied Science

Ottawa-Carleton Institute of Electrical Engineering
Faculty of Engineering
Department of Systems and Computer Engineering

Carleton University
Ottawa, Ontario, Canada K1S 5B6
August 28, 2002

@Copyright
2002, Fengjie Yuan

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

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

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

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

0-612-79722-8

Canada

The undersigned recommend to the Faculty of Graduate Studies and Research
Acceptance of the thesis

Performance Analysis of Resilient Packet Rings with Single Transit Buffer

submitted by

Fengjie Yuan, B. Eng, M. Eng.

in partial fulfillment of the requirements for
the degree of Master of Applied Science

Thesis Supervisor, Dr. Changcheng Huang

Chair, Department of Systems and Computer Engineering

Carleton University

August 28, 2002

Abstract

Resilient Packet Ring (RPR) is a new technology being standardized in the IEEE 802.17 working group. Several proposals have been made to the working group that optimize the performance and efficiency of packet traffic on ring topologies. In this thesis we will investigate the performance of one of the proposals, which features a single transit buffer, utilization-based congestion detection, and congestion notification with explicit rates. The key benefit of this proposal, known as 1TB-RPR (1 transit buffer RPR), is that it is extremely simple to implement in hardware. In addition to its simplicity, we will show that 1TB-RPR also performs well, and can be considered as an excellent candidate for IEEE 802.17 standard. The two key performance metrics for this ring technology are throughput (or utilization) and ring access delay. In this thesis we will show that 1TB-RPR can achieve a ring utilization as high as 95% of the ring capacity with a very low ring access delay.

Using a single transit buffer, 1TB-RPR accentuates the importance of ring access delay, especially for multiservice application. It is required that this delay be small and bounded so that all traffic streams can meet their end-to-end delay and delay jitter requirements. In this thesis, we develop a ring access delay bound for the 1TB-RPR system. The bound is then proved using both analytical and simulation approaches. Furthermore we show that the bound is tight by constructing a worst-case traffic scenario, in which the maximum ring access delay is equal to the bound.

Acknowledgements

I wish to thank my supervisor, Dr. Changcheng Huang, for his invaluable suggestions, guidance, and patience during the course of my thesis work. The constructive and timely suggestions offered in many discussions have brought forth a profitable exchange of ideas. His dedication and efforts are truly appreciated.

I would also like to thank Harry Peng, for his thoughtful advice. His enthusiasm and interest in the progress of this thesis have been inspiring.

Finally, I would like to thank my wife, Jingjing, for her support and love.

Table of Contents

Chapter 1 Introduction

1.1 Thesis Background and Motivation	1
1.2 Thesis objectives.....	4
1.3 Thesis organization.....	4
1.4 Thesis contribution.....	5
1.5 Publication.....	5

Chapter 2 background

2.1 SONET.....	7
2.2 Ethernet.....	7
2.3 Fiber Distributed Data Interface (FDDI), Distributed Queue Dual Bus (DQDB) and other related research.....	8
2.4 Resilient Packet Ring (RPR) feature overview.....	9
2.4.1 RPR Spatial Reuse.....	9
2.4.2 RPR Protection.....	10
2.4.3 RPR scalability.....	10

Chapter 3 The 1TB-RPR proposal

3.1 The 1TB-RPR node.....	12
3.2 Congestion span.....	14
3.3 State machine of the 1TB-RPR fairness algorithm.....	17

Chapter 4 The bound for 1TB-RPR ring access delay

4.1 Introduction.....	20
4.2 A bound for ring access delays in 1TB-RPR.....	22

4.3	A worst case for the RPR bound.....	25
Chapter 5 Simulation model and simulation results		
5.1	Simulation model.....	31
5.2	Simulation results for ring access delay bound.....	33
5.2.1	Case 1: Ring access delay with different leaky bucket size.....	36
5.2.2	Case 2: Ring access delay with different number of nodes on the ring.....	37
5.2.3	Case 3: Ring access delay with different token rate scenarios.....	38
5.2.4	Case 4: Ring access delay with different target utilizations.....	41
5.3	Simulation results for performance analysis of 1TB-RPR.....	43
5.3.1	Hub application.....	43
5.3.2	Distributed application.....	53
Chapter 6 Conclusions and future work		57
References		59

List of figures

Fig. 1 Typical RPR node.....	2
Fig. 2 A typical 1TB-RPR node.....	12
Fig. 3 An example of a congestion span.....	14
Fig. 4 An example of a congestion span with local traffic in the span.....	16
Fig. 5 1TB-RPR fairness state machine.....	17
Fig. 6 RPR vs. GPS.....	24
Fig. 7 A congestion span of N nodes.....	26
Fig. 8 Source traffic rates and tokens of the N nodes in the congestion span.....	27
Fig. 9 1TB-RPR node simulation model.....	31
Fig. 10 Ring access delay bounds and simulated max access delays under worst case traffic scenarios with different packet sizes.....	33
Fig. 11 Simulation network model set up.....	34
Fig. 12 Ring access delay bounds with different leaky bucket sizes.....	36
Fig. 13 Ring access delay bounds with different nodes on the ring.....	37
Fig. 14 Scenario 1 of weight distribution among the nodes on the ring.....	39
Fig. 15 Scenario 2 of weight distribution among the nodes on the ring.....	39
Fig. 16 Scenario 3 of weight distribution among the nodes on the ring.....	40
Fig. 17 Scenario 4 of weight distribution among the nodes on the ring.....	40
Fig. 18 Ring access delay bounds with different weight patterns.....	41
Fig. 19 Ring access delay bounds with different target utilizations.....	42
Fig. 20 Ring model for hub application.....	44

Fig. 21 Step response results.....	45
Fig. 22 Utilization of the link 1 to 0 in the step response scenario.....	46
Fig. 23 Piecewise linear results.....	47
Fig. 24 Utilization of the link 1 to 0 in the Piecewise linear scenario.....	48
Fig. 25 CDF of EF ring access delay of Node 1 at 95% target utilization and 100% load.....	49
Fig. 26 Average throughputs of all the nodes on the ring with 2000 bits bucket Size.....	51
Fig. 27 Average throughputs of all the nodes on the ring with 150k bits bucket Size.....	51
Fig. 28 1TB-RPR support for unequal bandwidth requirements.....	52
Fig. 29 Ring model for the distributed application.....	54
Fig. 30 Single queue vs. multiple (two) queues.....	56

List of Tables

Table 1 Common parameters set in the hub case.....	44
Table 2 Traffic configurations for the distributed application.....	55
Table 3 Theoretical throughputs for the distributed case.....	55

List of Acronyms Used

1TB-RPR	Resilient Packet Ring with 1 Transit Buffer
AF	Assured Forwarding (equal to Class B)
BE	Best Effort (equal to Class C)
BLSR	Bidirectional Line Switch Ring
CDF	Cumulative Distribution Function
EF	Expedited Forwarding (equal to Class A)
FDDI	Fiber Distributed Data Interface
GPS	Generalized Processor Sharing
GPS Bound	Bound for Ring Access Delay Calculated Using GPS Algorithm
HP	High Priority
IEEE	Institute of Electrical & Electronic Engineers, Inc.
LP	Low Priority
MAC	Media Access Control
MAN	Metropolitan Area Networks
RPR	Resilient Packet Ring
RPR Bound	Bound for Ring Access Delay Calculated in this thesis
TG	Traffic generator
VDQs	Virtual Destination Queues
WAN	Wide Area Network

Chapter 1 Introduction

1.1 Thesis Background and Motivation

Resilient Packet Ring (RPR) [1, 2] is a new data transport technology designed to meet the new requirements of Metropolitan Area Networks (MAN). Traditionally, MANs traffic has been carried over networks for voice traffic using Time Division Multiplexing (TDM). However, with the growth of Internet applications, network services have become increasingly data-centric. The traditional solution that maps data packets into circuit-switched timeslots is costly and inefficient from both the carrier and customer points of view. In order to optimize transport networks to handle the increase in data traffic, a new Media Access Control (MAC) layer protocol – Resilient Packet Ring (RPR) is under development. RPR shares SONET's ability to provide fast recovery from link and node failures, but also benefits from Ethernet's cost and simplicity by using statistical multiplexing. Furthermore RPR provides a fairness and congestion control mechanism that has not been addressed by either SONET or Ethernet, which allows RPR to be significantly more efficient than all previous MAN technologies. Several vendors are shipping and continue to develop RPR products. The Institute of Electrical & Electronic Engineers, Inc. (IEEE) has set up the 802.17 RPR working group to standardize this technology.

An RPR network is a ring-based architecture that consists of two counter-rotating rings with each station connecting to two adjacent stations over a link pair. RPR has also generalized the spatial reuse concept that can be found in SONET BLSR to allow low priority traffic to use the redundant capacity otherwise reserved for protection purposes. A typical RPR

node structure is shown in Fig 1. An RPR node has three kinds of traffic, namely add-in traffic, drop-off traffic and pass-through traffic. It is well known that statistical multiplexing can increase the overall efficiency of a transport system by accommodating multiple bursty traffic streams. The key difference between SONET/SDH and RPR is that an RPR node employs statistical multiplexing at the packet level. As shown in Fig. 1 the add-in traffic stream is statistically multiplexed with the pass-through traffic stream yielding a higher utilization of the fiber link.

Fig. 1 Typical RPR node

Since add-in traffic may compete with pass-through traffic for access to the ring's bandwidth when an outgoing link becomes congested, buffers are needed to avoid temporal contention or packet collision. A RPR node uses the so-called transmit buffer and transit buffer

to hold the bursts of add-in traffic and pass-through traffic respectively when congestion happens. Both transmit buffer and transit buffer can be further partitioned to hold traffic flows with different priorities, e.g., one for high priority (HP) traffic and one for low priority (LP) traffic. A scheduler is required to decide which traffic stream can use the bandwidth at a particular moment. When the pass-through traffic flows are given priority over the add-in traffic flows, add-in traffic flows will be temporally blocked and therefore suffer ring access delay. This is analogous to a collision in Ethernet terms. But in the case of RPR, a simple yet intelligent bandwidth and congestion management scheme within MAC results in significantly more efficiency than Ethernet. Several proposals have been made to the IEEE 802.17 WG on how the bandwidth and congestion management scheme should be implemented. The proposals differ in the number of transit buffers being implemented, and the way congestion is detected and controlled [3, 4, 5]. In this thesis we will investigate the performance of one of the proposals originally proposed by Nortel [6]. This proposal, known as 1TB-RPR, features single transit buffer, utilization-based congestion detection, and congestion notification with explicit rates.

Compared to other proposals that use multiple transit buffers, 1TB-RPR is extremely simple to implement in hardware. Under the 1TB-RPR proposal, pass-through packets always have priority over add-in packets. The only chance that a pass-through packet can be blocked is when an add-in packet is being transmitted. Therefore the size of the transit buffer can be set as small as the max packet size. Under the multiple transit buffers proposals, the high priority add-in packets can block the low priority pass through packets. To hold the blocked low priority packets on the ring, much larger buffers are needed compared to 1TB-RPR.

Considering the high link rate, multiple transit buffers proposals are much more costly than 1TB-RPR from implementation point of view.

Since pass-through packets always have priority over add-in packets under the 1TB-RPR proposal, an add-in packet arriving from the ingress port will be blocked and therefore suffer a ring access delay no matter what kind of priority it has, if there is pass-through traffic on the ring at the same time.

In this thesis, we will investigate the two key performance metrics for a ring technology, which are throughput (or utilization) and ring access delay. We will show that 1TB-RPR can achieve as high as 95% utilization with a small ring access delay. Furthermore, we will prove that there is a bound for ring access delay in 1TB-RPR.

1.2 Thesis objectives

This thesis has two main objectives. One objective is to prove that there exists a bound for ring access delay in a 1TB-RPR system. We also prove that the bound we developed for ring access delay is tight by constructing a worst-case traffic scenario, in which the maximum ring access delay is equal to the bound. The other objective of this thesis is to investigate the performance of 1TB-RPR by simulation to prove that 1TB-RPR is an excellent candidate for 802.17 standard. In performance analysis for 1TB-RPR, we mainly focus on the two key metrics, which are utilization and ring access delay.

1.3 Thesis organization

The thesis is organized as follows:

Chapter 2 describes the background information for the thesis including the main features of RPR and comparison RPR to other related technologies.

Chapter 3 introduces the 1TB-RPR proposal, which features single transit buffer, utilization-based congestion detection, and congestion notification with explicit rates. This chapter also describes the issues to be investigated in the thesis.

Chapter 4 focuses on the research on the bound for 1TB-RPR ring access delay. In this chapter we develop the bound for the ring access delay and prove that it is tight by constructing a worst-case traffic scenario, in which the maximum ring access delay is equal to the bound.

Chapter 5 presents simulation results for both the ring access delay bound and the performance analysis for 1TB-RPR. In this chapter we examine the ring access delay, utilization, and fairness issues through simulations.

Chapter 6 concludes the thesis.

1.4 Thesis contribution

The following are the main contribution of this thesis:

- Developed a bound for ring access delay of 1TB-RPR.
- Created a simulation model package based on the description of 1TB-RPR proposal, and provided recommendation to IEEE 802.17 Resilient Packet Ring Working Group with the performance analysis of 1TB-RPR. The simulation model package has been released by Nortel as a reference model on the web site of IEEE 802.17 Resilient Packet Ring Working Group.

1.5 Publication

- [1] Fengjie Yuan, Changcheng Huang, Harry Peng, John Hawkins, "Performance Analysis for Resilient Packet Rings with Single Transit Buffer," *IEEE ICT 2002*, Beijing China, June 2002.
- [2] Fengjie Yuan, Changcheng Huang, Harry Peng, John Hawkins, "A bound for Ring Access Delays of Resilient Packet Rings with Single Transit Buffer," Submitted to *IEEE INFOCOM 2003*.

Chapter 2 Background

This section first discusses some related algorithms and their limitations in Metro Rings. Then it introduces the general features of Resilient Packet Ring.

2.1 SONET

The existing Metro network infrastructure is SONET-based network. As mentioned in the last section, there are well-known disadvantages in using SONET for transporting data traffic. First, SONET provisions point-to-point circuits between ring nodes and each circuit is allocated a fixed amount of bandwidth [7][8]. The fixed bandwidth allocation on a SONET ring limits the maximum burst traffic between endpoints, though the data traffic is inherently bursty. In addition, provision the circuits on a SONET ring results in extremely inefficient use of ring bandwidth when the network is designed for a logical mesh [9][10]. Second, SONET ring wastes bandwidth in handling multicast traffic. On a SONET ring, multicast traffic requires that a separate copy of the packet be sent to each destination [11], so multiple copies of multiple packets travel around the ring, wasting bandwidth. Third, SONET wastes protection bandwidth in that 50 percent of bandwidth is reserved for protection [12][13]. While protection is obvious important, SONET does not achieve this goal in an efficient way.

2.2 Ethernet

With scalability of service interface, Ethernet is well known as an inexpensive solution for data traffic. Gigabit Ethernet is also considered a big step in the evolution of packet-based transport in the MAN [14][15]. But Ethernet is not suitable for MANs for the following reasons. First, most of the existing MANs are ring-based networks, in which rings act as a shared media and need media access control (MAC) mechanisms to manage access across multiple users. Ethernet does not work efficiently in media access control and therefore cannot achieve high utilization in ring configurations [16]. Second, Ethernet does not take advantage of a ring topology to implement a fast protection mechanism. Instead, Ethernet achieves protection by path redundancy utilizing a spanning tree protocol, which recovers comparatively slowly from a fiber cut because the recovery mechanism requires the failure condition to be propagated serially to each upstream node [17]. Third, though Ethernet can provide link-level fairness, it is not good at implementing global fairness policies for sharing ring bandwidth [18][19], which is very important in MANs.

2.3 Fiber Distributed Data Interface (FDDI), Distributed Queue Dual Bus (DQDB) and other related research

The reason why we mention FDDI [20][21] and DQDB [22][23] here is that both protocols can be employed in ring based networks. But FDDI has a problem in scalability because it uses tokens passing around a ring to manage the ring access. In addition, FDDI does not permit spatial reuse and concurrent transmission, since at most one node that holds the token can transmit into the ring at any given time. While FDDI is not comparable to RPR, how about DQDB in MAN? DQDB does offer lower access delay than FDDI [24], but it depends

on a round robin algorithm to serve the downstream nodes [25]. Therefore DQDB does not efficiently use the ring bandwidth and also has a problem with scalability.

There are many other ring-based protocol-proposals investigated by researchers, such as MetaRing [26][27]. Maintaining quota-based fairness with spatial bandwidth reuse for data traffic, MetaRing did not find its way to the market as a local-area network [27]. However, the MetaRing's SAT fairness algorithm [28] has been implemented in serial storage architectures, such as disk drives, tape drives etc.

2.4 Resilient Packet Ring (RPR) features overview

RPR addresses the current and future trends in MAN and WAN networking with several attractive characteristics described as follows.

2.4.1 RPR Spatial Reuse

The RPR protocol uses spatial reuse to increase the overall efficiency of the ring. Different from earlier ring-based protocols, such as token ring [29] and fiber distributed data interface (FDDI) protocols, in which traffic was removed from the ring by the source node, RPR employs destination stripping technology in which the destination node strips its packets from the ring. Destination stripping allows traffic to be passed bi-directionally on the ring along the spans between source and destination nodes without using the bandwidth over other segments of the ring, and therefore achieves spatial reuse. Spatial reuse mechanism in metropolitan rings is extremely advantageous because it improves bandwidth efficiency greatly and increases the effective bandwidth available on the ring.

2.4.2 RPR Protection

RPR offers service protection by performance monitoring and rapid self-healing. Unlike traditional SONET networks, RPR does not need dedicated protection bandwidth. Instead, RPR uses both rings to carry working traffic. The RPR network protects and re-establishes connectivity automatically, with protection message exchanged quickly to inform other nodes of the status of the network. As the RPR standard has not been finished, a well-defined protection mechanism is needed for plug-and play operation of RPR nodes in order to allow the insertion and removal of nodes without provisioning.

2.4.3 RPR scalability

In current MAN, it takes a long time for carriers to provision services, because the provisioning operation in the underlying SONET infrastructure is time and labor consuming. To provision a new service, the network operator needs to do a lot of traffic engineering manually to optimize bandwidth utilization on the ring, and must be aware of the network topology, the traffic distribution on the ring and the available bandwidth on each logical link.

In contrast, the RPR approach is more scalable in that it functions as a shared medium. In a RPR system, all nodes share bandwidth on the ring. To provision a new service the network operator simply identifies a traffic flow and specifies the quality of service that it should get in the system, instead of a node-by-node and link-by-link capacity planning. In addition, a RPR system simplifies service upgrades significantly, especially if the subscriber is connected to the RPR system via a scalable interface such as Ethernet.

RPR has the potential to change the way in which data services are transported in the MAN and WAN. It transforms the underlying infrastructure for data delivery from static to dynamic. This opens up the possibility for a new set of dynamic data service in MAN.

Chapter 3 The 1TB-RPR proposal

In this chapter we introduces the 1TB-RPR proposal, which features a single transit buffer, utilization-based congestion detection, and congestion notification with explicit rates. We also describe the issues to be investigated in the thesis.

3.1 The 1TB-RPR node

Fig. 2 A typical 1TB-RPR node

A typical 1TB-RPR node is shown in Fig. 2. The 1TB-RPR scheduling algorithm is based on Buffer Insertion Ring (BIR) technology [30] where pass-through packets always have priority over add-in packets from the transmit buffers. When the transit buffer has traffic to send, add-in packets will be queued in the transmit buffer until the transit buffer is emptied. Because the pass-through traffic has absolute priority over the add-in traffic, only a very small

transit buffer (2 or 3 packet sizes) is required. This significantly simplifies the hardware implementation of the MAC. But on the other hand all add-in traffic streams ring access delays will be affected. For high priority traffic (Class A and Class B) this ring access delay contributes a delay jitter that must be minimized. To reduce the ring access delay, a fairness algorithm based on feedback control is designed to control the access of the total bandwidth for all nodes during periods of congestion. To detect congestion, the fairness algorithm uses two trigger conditions: one triggered by a high utilization, and one triggered by a high ring access delay. When congestion is detected, the 1TB-RPR-fairness algorithm uses explicit congestion notifications to manage bandwidth on the ring so that weighted bandwidth fairness is achieved and ring access delays are minimized. The weight assigned to a node represents how much bandwidth the node requires for low priority traffic during periods of congestion. To support weighted fairness, each station weight can be distributed by a topology discovery mechanism of RPR. This topology discovery method is outside the scope of this paper.

The utilization is estimated using a weighted sliding-window estimator. The equation for the estimated throughput is as follows:

$$ESTIMATEDRate(t) = ESTIMATEDRate(t-1) - \frac{ESTIMATEDRate(t-1)}{WEIGHT1} + \frac{CURRENTrate}{WEIGHT2}$$

where $ESTIMATEDRate(t)$ is the current estimated “rate” in bytes, $WEIGHT$ is an integer value of: 2, 4, 8, 16,128, and $CURRENTrate$ is a sampled rate measured in bytes. It should be noted that $WEIGHT$ here decides how fast the $ESTIMATEDRate(t)$ is updated and is totally different from the weights assigned to each node for fairness purposes.

Fig. 2 shows a typical 1TB-RPR-node structure in which the scheduler chooses data packets from five queues:

- Packets from transit buffer.

- Packets from Class A transmit buffer.
- Packets from Class B transmit buffer.
- Packets from in-span Class C transmit buffer.
- Packets from out-of-span Class C transmit buffer.

Class A and Class B traffic is engineered according to a Committed Information Rate (CIR) and is not subject to the advertised rates generated by the fairness algorithm in downstream nodes. A Class C traffic flow is regulated by a token bucket where its token rate is controlled by the advertised rate received from the downstream nodes during periods of congestion and its bucket size is pre-configured to achieve the maximum throughput.

Fig. 3 An example of a congestion span

3.2 Congestion span

Fig. 3 gives an example of a congestion span, which is defined as the span of all nodes contributing to the congestion on a link. A congestion span typically consists of a head node, several chain nodes and a tail node. In Fig. 3, for example, node 4 is a head node, node 2 and node 3 are chain nodes and node 1 is a tail node. A node that detects a congested outgoing link is defined as the head node. The head node knows the whole congestion span because each node tracks the IDs of all source nodes with traffic passing through them. Based on the utilization of its outgoing link, which is also called a downstream link, the head node calculates a fair rate for Class C traffic and then advertises it to the upstream nodes, the nodes sending traffic passing the head node. To speed up convergence, the initial advertised rate is normalized by the sum of all the weights assigned to the nodes within the congestion span; this provides faster convergence. Having received the normalized advertised rate from the downstream node, each node calculates its target rate by multiplying the normalized advertised rate with its own weight and then applying the rate to its leaky bucket for the out-of-span Class C traffic. Assume there are N nodes (node 1 – node N) in a congestion span. Let ρ_i be the token rate of the leaky bucket in node i , ω_i be the weight assigned to node i , U_T be the target utilization, C be the link speed and C_H be the mean rate of high priority traffic (Class A and B) on the outgoing link of node N , then we will have

$$\rho_i = \frac{\omega_i}{\sum_{j=1}^N \omega_j} (U_T C - C_H). \quad (1)$$

Using this scheme the 1TB-RPR-fairness algorithm distributes any spare capacity to all the nodes in the congestion span in a weighted fashion.

Fig. 4 An example of a congestion span with local traffic in the span

The example shown in Fig. 3 is a hub application where each node sends traffic to a hub node. Another important ring application is the distributed case where each node sends traffic to any other node on the ring. In the distributed case, the congestion control scheme avoids adversely affecting the local traffic that does not contribute to the congestion on the congested link as shown in Fig. 4. Otherwise the 1TB-RPR-fairness algorithm would be unfair and would result in loss of efficiency. The solution to this issue is to use Virtual Destination Queues (VDQ) that can be implemented outside the MAC layer. Packets are queued in different buffers based on their destinations. The leaky bucket only applies the advertised rate to the packets whose destinations are out of the congestion span. At least two VDQs are required (as shown in Fig. 2), one in-span queue that buffers the in-span traffic flows and one out-of-span queue that buffers out-of-span traffic flows.

3.3 State machine of 1TB-RPR fairness algorithm

To let readers better understand 1TB-RPR fairness algorithm, we introduce the state machine that is the heart of the fairness scheme. As shown in Fig. 5, the fairness state machine consists of 4 major states and 4 minor states. The minor states automatically exit after the required operation is completed. The major states are described as follows:

Fig. 5 1TB-RPR fairness state machine

- Normal or Un-Congested state

This is the default state when a RPR node is powered on.

- HEAD 1 state

When a node in Normal state detects congestion, the node will change its state from Normal to HEAD 1. When congestion is released, the node will go back to Normal state if it is no longer in a congestion span. Otherwise it will change to CHAIN state.

While in HEAD 1 state, the node advertises the normalized rate to the upstream nodes and also calculates its own target rate applied it to its add-in traffic. Meanwhile, the node in HEAD 1 state continuously adjusts its advertised rate up or down depending on the output link utilization.

- * Head Up state: This is a minor state to lower the advertised rate to the next lower value.

- * Head Down state: This is a minor state to raise the advertised rate to the next high value.

- HEAD 2 state

If a node in CHAIN state also detects congestion in its outgoing link, the node will change its state to HEAD 2. This second head state is required to take into account congestion that may occur within the span. As the traffic pattern changes, not all traffic flows from the tail to the head. Congestion may occur within the span or multiple spans may merge into one. The HEAD 2 state allows a single span to encompass other spans. HEAD 1 and HEAD 2 differ in their advertised rate calculations. The node in HEAD 1 state is the true head node of the congested span. It controls the target rate for all nodes in the span. The node in HEAD 2 state will propagate the minimum target rate received from the node in HEAD 1 state or from its own perspective.

- CHAIN/TAIL state

A node can jump into CHAIN state from Normal state, HEAD 1 state or HEAD 2 state when it is in a congestion span but experiences no congestion on its outgoing link. A node in CHAIN state is identified as a node that receives a target rate lower than the maximum

rate allowed. A chain node forwards the received target rate upstream while applying it to its leaky bucket. The tail state differs from a chain state in that it has very little pass-through traffic. A tail node does not forward its downstream rate to its upstream node, but simply applies the received rate to its own leaky bucket. The node will change back to Normal state from CHAIN or TAIL state if the congestion span is removed.

A simulation model using OPNET was built to investigate the performance of 1TB-RPR. In the chapter 5 we will discuss the simulation results.

Chapter 4 A bound for ring access delays of 1TB-RPR

In this chapter, we develop the bound for the ring access delay and prove that it is tight by constructing a worst-case traffic scenario, in which the maximum ring access delay is equal to the bound.

4.1 Introduction

Because the pass-through traffic has absolute priority over the add-in traffic, only a very small transit buffer (2 or 3 packet sizes) is required. This significantly simplifies the hardware implementation of the MAC. But on the other hand all add-in traffic streams' ring access delay will be affected. For high priority traffic (Class A and Class B) this ring access delay contributes a delay jitter that must be minimized. It is clear that ring access delays are the same for all kinds of add-in packets, so the bound for ring access delays of low priority add-in traffic is also the bound for that of high priority add-in traffic. Because the ring access delay is the waiting time of an add-in packet when it is blocked by pass-through traffic, so it is equal to the burst of pass-through traffic on the ring. Typically Class A and B are tightly shaped according to CIRs and therefore are not the main contributors to the ring access delays of downstream nodes. Since Class C is not shaped according to CIRs, Class C contributes the bursts on the ring. Therefore in the following sections we will assume that there is only Class C traffic on the ring without losing generality (i. e. $C_H = 0$).

This analysis also applies to the multiple transit buffer design for the same priority packets in transit competing with the add-in traffic.

Delay bounds for various systems have been extensively studied by many researchers [33 - 41]. From [42] we can see that Ethernet does not have a bound for access delay. Ethernet uses an exponential backoff algorithm to control the access to the link, so a packet that suffered a collision last time will not be guaranteed to access the link next time. Karvelas and Leon-Garcia [43] developed a bound for the access delay of a packet at its transmitter station in a FDDI ring. The bound can be explained as follows:

Let $W_{\max}$ be the maximum waiting time for a packet, V_s be the number of traffic sources on the ring, and h_v be the packet transmit time, we have

$$W_{\max} = T_{TTP} + (V_s - 1)h_v,$$

where T_{TTP} is the parameter to control the available time for data transmission.

Cidon and Georgiadis [41] presented a bound for Head of Line delay (HOL) of MetaRing. The bound can be explained as follows:

HOL Delay bound for Algorithm A of SAT [27]:

$$6\frac{Q_{\max}}{B_w} + 3T_{RD} + 2n\frac{L_{\max}}{B_w}$$

HOL Delay bound for Algorithm B of SAT:

$$4(\ln n + 1)\frac{Q_{\max}}{B_w} + 3T_{RD} + 2n\frac{L_{\max}}{B_w}$$

where n stands for the number of the nodes on the ring, $Q_{\max}$ is the maximum quota granted by the SAT, T_{RD} is the round-trip delay, $L_{\max}$ is the maximum size of a packet, B_w is the link bandwidth. The HOL delay bound studied by Cidon is for quota or slot based fairness,

therefore it is different from the ring access delay bound of the 1TB-RPR that is controlled by a rate based fairness algorithm.

The research methods of Generalized Processor Sharing (GPS) System [33][34], which are very useful to our study, will be introduced in detail in the next section. Yu [35] developed a bound for the individual session queue length, which is useful for Quality of Service control in a GPS system. Diwan [36] derived a way to provide end-to-end delay guarantees under PGPS Scheduling. Sivaraman [37] focused on the relationship between traffic shaping and end to end delay with EDF Scheduling. Valaee [38] devised a recursive algorithm to estimate the worst-case burstiness in a connection-oriented network. From the work of Andrews [39] and Starobinski [40] we can see some research methods for studying stochastic delay bound for systems.

Since RPR is a brand-new concept, it has been rarely studied. We believe that this paper is one of the earliest research works to be conducted on the ring access delays of a RPR system.

In our research on 1TB-RPR bound for ring access delay, we develop a bound for ring access delays in 4.2. We then construct a worst case to prove that the bound is tight in 4.3. In 5.2, we present some simulation results about ring access delays under different parameters.

4.2. A bound for ring access delays in 1TB-RPR

Cruz [31][32], Parekh and Gallager [33][34] have given a theoretical bound for traffic bursts in a GPS (Generalized Processor Sharing) system. A GPS system has N sessions sharing a common out-going link under an arbitrary work-conserving scheduler as shown in the top

part of Fig. 6. Each of the N user sessions is regulated by an ingress leaky bucket (σ_i, ρ_i) .

The GPS bound, the bound of the maximum traffic burst on the outgoing link, will be [2]

$$G_N = \frac{\sum_{i=1}^N \sigma_i}{C - \sum_{i=1}^N \rho_i} \quad (2)$$

where C is the outgoing link capacity.

Since the ring access delay of a node in a RPR ring equals the pass-through burst that the node sees, it is clear that the maximum ring access delay happens when all the bursts concatenate together when they pass-through the node as shown in the last section. Once a packet is on the ring it suffers very little delay jitter while propagation delays from node to node are constant. Therefore the congestion span shown in Fig. 3 can be modeled as the lower part of Fig. 6. Comparing the two diagrams in Fig. 6, we can say that a 1TB-RPR congestion span is different from a GPS system in that a 1TB-RPR congestion span does not have buffers between leaky buckets and the RPR scheduler. (This assumes the spatial separations between stations are small and there is little packet buffering in the fiber.)

Fig. 6 RPR vs. GPS

Because both RPR and GPS are work conserving and constrained by leaky bucket (σ_i, ρ_i) , equation (2) can be applied to ring access delays in 1TB-RPR as proved in the following Lemma. We therefore define this bound as the RPR bound for ring access delays.

Lemma 1: With the same set of parameters that satisfy equation (1), the GPS bound in equation (2) is applicable to a congestion span of an 1TB-RPR system.

Proof:

The constraints imposed by the leaky bucket in node i are as follows: If $A_i(\tau, t)$ is the amount of flow that leaves the leaky bucket and enters the ring in time interval $(\tau, t]$, then

$$A_i(\tau, t) \leq \sigma_i + \rho_i(t - \tau), \forall t \geq \tau \geq 0. \quad (3)$$

Define a burst to be a interval B such that for any $\tau, t \in B, \tau \leq t$,

$$\sum_{i=1}^N A_i(\tau, t) = (t - \tau)C \quad (4)$$

If $B=[t_1, t_2]$, from (3)(4), we have

$$B = t_2 - t_1 \leq \frac{\sum_{i=1}^N \sigma_i}{C - \sum_{i=1}^N \rho_i} \quad (5)$$

In equation (5), $\sum_{i=1}^N \sigma_i$ is the maximum number of tokens that a congestion span can accumulate, C is the token consumption rate while the outgoing link of Node N is busy, and $\sum_{i=1}^N \rho_i$ is the total token replenishing rate. It is easy to see now that the bound stated in equation (5) is a bound imposed by the over-all token supply. That is why it works for both GPS systems and RPR systems. But as discussed in the next section, the worst cases that achieve this bound are completely different for the two systems.

Fig. 7 A congestion span of N nodes

4.3. A worst case for the RPR bound

In this section we construct a worst traffic scenario for a set of arbitrary parameters that satisfy equation (1) to show that the RPR bound of equation (5) is a tight bound for ring access delays. In general we assume there are N nodes in a congestion span as shown in Fig. 7, with (σ_i, ρ_i) constraints. To simplify the case, we assume $C_H = 0$, because only Class C contributes the bursts on the ring.

Lemma 2: The RPR bound in equation (5) is tight for ring access delays of an 1TB-RPR system.

Proof:

We use a constructive approach to prove that the bound is tight. We will show that we can always find a worst traffic case in which the maximum ring access delay equals to the RPR bound for a set of arbitrary parameters that satisfy equation (1).

Fig. 8 Source traffic rates and tokens of the N nodes in the congestion span

Parekh and Gallager [33] employ greedy sessions, sessions that use as many tokens as possible, to prove that the bound as calculated by equation (2) for the output traffic bursts of a GPS multiplexing system is tight. It is shown in [33] that when all the sessions are greedy at the same time, the GPS bound can be achieved. Although it is intuitively natural to consider concurrent greedy sessions as the worst traffic scenario for GPS systems, it is not the case for an RPR system. This is because a 1TB-RPR congestion span does not have any buffer between its leaky buckets and its scheduler. In an RPR congestion span, upstream nodes have priority over down stream nodes, so the down stream nodes will lose tokens when they are blocked by the traffic from their upstream nodes if their buckets are full. From equation (5) we can see that it will make the burst shorter if any token is lost. Therefore concurrent greedy sessions will not be the worst case in an RPR system. In the following we will construct a worst case traffic scenario for an RPR system.

Fig. 8 is our worst-case traffic loading scenario. As shown in Fig. 8, we assume that buckets are full at t_0 . The source traffic rate of node 1 is set to $C - \sum_{i=2}^N \rho_i$, while the source traffic rates of node 2 to node N are set to $\rho_2, \rho_3 \dots \rho_N$. Therefore the total amount of traffic the N nodes add on to the ring is

$$C - \sum_{i=2}^N \rho_i + \rho_2 + \rho_3 + \dots + \rho_N = C.$$

That is to say that the ring is busy from t_0 . It should be noted that at this stage the number of tokens in the node 1 bucket will decrease while all other buckets maintain the same amount of tokens if equation (1) is satisfied. We set the source traffic rate of node 1 at $C - \sum_{i=2}^N \rho_i$ until the

leaky bucket in node 1 runs out of tokens at t_1 . It is easy to see that the busy period $[t_0, t_1]$ will be

$$B_1 = \frac{\sigma_1}{C - \sum_{i=2}^N \rho_i - \rho_1} = \frac{\sigma_1}{C - \sum_{i=1}^N \rho_i}.$$

As shown in Fig. 8, when node 1 runs out of tokens at t_1 , we set the source traffic rate of node 1 to ρ_1 . At the same time, we set the source traffic rate of node 2 to $C - \rho_1 - \sum_{i=3}^N \rho_i$, and keep this value until the leaky bucket in node 2 runs out of tokens at t_2 while all other nodes stay at their original rates. This busy period $[t_1, t_2]$ will be

$$B_2 = \frac{\sigma_2}{C - \rho_1 - \sum_{i=3}^N \rho_i - \rho_2} = \frac{\sigma_2}{C - \sum_{i=1}^N \rho_i}.$$

When there are no tokens left in the leaky bucket of node 2 at t_2 , the source traffic rate of node 2 goes back to ρ_2 , and the source traffic rate of node 3 goes up to $C - \sum_{j=1}^2 \rho_j - \sum_{i=4}^N \rho_i$.

We can repeat this process until we finish all the nodes in a congestion span as shown in Fig.

8. Therefore the total burst length will be

$$B = B_1 + B_2 + \dots + B_N = \frac{\sigma_1}{C - \sum_{i=1}^N \rho_i} + \frac{\sigma_2}{C - \sum_{i=1}^N \rho_i} + \dots + \frac{\sigma_N}{C - \sum_{i=1}^N \rho_i} = \frac{\sum_{i=1}^N \sigma_i}{C - \sum_{i=1}^N \rho_i} \quad (6)$$

From (6) we can see that the maximum burst length in this specific case is exactly equal to the bound for ring access delays as defined in (5). By this case we have proved that

the bound stated in equation (5) for ring access delays in 1TB-RPR is tight.

From Fig. 8 we can see that Node 1 is not greedy until t_1 , Node 2 is not greedy until t_2 and so on and so forth. This is very different from the worst-case scenario in [2] where all the sessions are greedy from time t_0 .

Chapter 5 Simulation model and simulation results

In this chapter we first introduce the model we built using OPNET to investigate the performance of 1TB-RPR. We then examine the simulation results regarding the bound for ring access delay. In this part we ran simulations with different parameters affecting the ring access delay, and compared the max ring access delay to both GPS bound and RPR bound. Finally, we examine the simulation results for the performance analysis of 1TB-RPR in both a hub application and a distributed application.

Fig. 9 1TB-RPR node simulation model

5.1 Simulation model

Fig. 9 shows the architecture of the node mode implemented in OPNET. The node architecture is symmetric around its center. The node is composed of ``modules" that are interconnected to one another by arrowed lines. There are two types of arrowed lines: solid lines representing ``packet streams", and dashed lines representing ``statistic wires". Arrows indicate the flow directions of packets or statistic information among the modules. Each module is a ``process model" in OPNET. The node supports two directions of connection, which are indicated by ``1" or ``2" in the module names. It should be noted that the modules from one direction interact with the modules from the other direction. We group the modules according to their functionalities in packet transmission and fairness information processing. There are mainly four function blocks in the node structure, which are ring-input block, ring-output block, local ingress/egress block, and fairness block.

Ring-input block includes a "ring in" module and a "Msg_dispatcher" module. The Ring-input block receives packets from the ring and checks whether they are local traffic, pass-through traffic, or a fairness message.

A Ring-output block consists of three modules, which are a "Ring out" module, a "Scheduler" module, and a "PT_FIFO" module. The main function of this block is to schedule the packets from different streams and send them out onto the ring.

The modules in the local ingress/egress block are a "MAC IN" module, a "MAC OUT" module, a "MAC ingress/egress" module, and a "Leaky bucket" module. This block gives back pressure to queues outside the MAC layer by exchanging information with the queues. This block also applies fair rate control to local traffic onto the ring.

A Fairness block has 4 modules, which are a “Fairness_cal” module, a “Fairness_msg_gen” module, a “Clock_gen” module, and a “Ctl_FIFO” module. This block implements the 1TB-RPR-fairness algorithm, which is described in Chapter 3.

5.2 Simulation results for ring access delay bound

In the proof of Lemma 1 and 2 it was assumed that traffic followed a fluid model. In practice all traffic streams are packetized. To check the impact of packetization, we have

Fig. 10 Ring access delay bounds and simulated max access delays under worst case traffic scenarios with different packet sizes.

conducted several simulations with 8 nodes as a congestion span. In figure 10, the pair of dotted lines are the calculated bounds and the simulated maximum access delays under the worst case traffic scenario as identified in Lemma 2 with packet size of 12272 bits. The pair of solid lines are the results with packet size of 4224 bits. The reason that the calculated bounds are different with different packet sizes is because 1TB-RPR allows their leaky buckets to have a maximum deficit of 1 packet size. It is easy to see that there are errors between calculated and simulated results. But the errors become smaller when the packet size becomes smaller. Therefore it can be concluded that these errors are introduced by packetization. In general they are small and negligible.

Fig. 11 Simulation network model set up

From the analysis in Chapter 4, we can see that the bound on ring access delay is related to four factors: the number of nodes on the ring, each node's leaky bucket size, the token rate of the leaky bucket in each node, and the target utilization of the ring. In this section, we compare simulation results with their analytical bounds regarding these four factors.

We have developed a 1TB-RPR model using OPNET and studied the hub application where all nodes send traffic to the hub node only because it is the worst-case scenario in terms of congestion and ring access delays (the whole ring becomes a congestion span). The ring set up is shown in Fig. 11.

The traffic for the simulation is described as follow:

- The packet inter-arrival distribution is exponential (Poisson traffic).
- The packet size distribution is trimodal (60% 64 byte, 20% 512 byte, 20% 1518 byte).
- The mean packet size is 444.4 bytes.

Since the bound on ring access delays is related to four factors, we run four cases varying parameters corresponding to each factor. The purpose of Case 1 is to check the bound with regard to the leaky bucket size. Case 2 shows the bound with different numbers of nodes on the ring. Case 3 presents the bound of ring access delay with different token rates. Case 4 gives the relation between the bound of ring access delay and target utilization, which is defined in a ring with N nodes as

$$U_t = \frac{\sum_{i=1}^N \rho_i}{C}.$$

The common parameter in the four cases simulation are described as follow:

- Ring link rate is 10Gbps.

- Propagation delay per link is $70 \mu s$.

All nodes are loaded with concurrent greedy sessions. Statistics are collected after steady states are established.

5.2.1 Case 1: Ring access delay with different leaky bucket sizes

Fig. 12 Ring access delay bounds with different leaky bucket sizes

In this case, there are 8 nodes on the ring. Each node's weight is equal so that the token rate of the leaky bucket in each node is equal. The target utilization of the ring is 0.95. Fig. 12 shows the comparison of the RPR ring access delay bound and the max simulated ring access delay with different leaky bucket sizes. From Fig. 12 we can see that the ring access delay bound becomes higher when the nodes on the ring use larger bucket sizes. It can also be seen

that the simulated max delay under concurrent greedy sessions are far from the worst-case access delays as shown by the bounds. This becomes more trivial when the bucket sizes become larger and larger.

5.2.2 Case 2: Ring access delay with different numbers of nodes on the ring

Fig. 13 Ring access delay bounds with different numbers of nodes on the ring

Setting equal weights for all nodes, 6000 bits bucket size for each node, and 0.95 target utilization, we run a series of simulations with different numbers of nodes on the ring. The measured max ring access delays in this case and the related analytical bounds are shown in Fig. 13, from which we can see that the more nodes on the ring, the larger the ring access delay

bounds. This is easy to see because with a fixed leaky bucket size, adding more nodes on to the ring means adding more buckets onto the ring. The errors between calculated bounds and max simulated access delays under concurrent greedy sessions become larger when there are more nodes contributing to the congestion.

5.2.3 Case 3: Ring access delay with different token rate scenarios

In this case, we use 16 nodes on the ring, and set the leaky bucket size to 6000 bits for each node and use 0.95 target utilization. With each node on the ring sending traffic to the hub, the ring access delay is measured in node 15, which is the nearest node to the hub. It should be noted that node 1 is the farthest away from the hub. As mentioned in Chapter 3, when congestion occurs, the nodes within the congestion span will control the low priority traffic on to the ring according to the available bandwidth on the ring and each node bandwidth requirement, which is indicated by weight. The node with higher weight has higher token rate when congestion happens.

In order to test the reaction of ring access delay to different token rate, we design four weight scenarios as shown in Fig. 14 (Scenario 1), Fig. 15 (Scenario 2), Fig. 16 (Scenario 3) and Fig. 17 (Scenario 4). In scenario 1, node 15 is allocated with the maximum bandwidth requirement while node 1 is allocated with minimum bandwidth requirement. From node 1 to node 15, the bandwidth requirement grows linearly. In scenario 2, the bandwidth requirement grows from node 1 to node 8 linearly, and descends from node 8 to node 15 linearly, with node 8 requiring the maximum bandwidth. The bandwidth requirements of the entire nodes on the ring are equal in scenario 3. Scenario 4 is exactly the reverse of scenario 1. From results shown in Fig. 18, we can see an interesting outcome. Both the calculated bound and the maximum

Fig. 14 Scenario 1 of weight distribution among the nodes on the ring

Fig. 15 Scenario 2 of weight distribution among the nodes on the ring

Fig. 16 Scenario 3 of weight distribution among the nodes on the ring

Fig. 17 Scenario 4 of weight distribution among the nodes on the ring

Fig. 18 Ring access delay bounds with different weight patterns

simulated ring access delays in scenario 1 are relatively smaller than those in all three other scenarios. In scenario 1, node 15 requires the max bandwidth when congestion happens. Therefore the total amount of tokens collected by nodes 1 to node 14 is smaller than that in other scenarios. For this reason, the calculated bound and simulated delays are the smallest for scenario 1.

5.2.4 Case 4: Ring access delay with different target utilizations

In this hub case, there are 8 nodes on the ring as shown in Fig. 11. The weights of all nodes are equal. The bucket size of each node is 6000 bits. Fig. 19 illustrates the RPR bounds

and the max simulated ring access delays with different target utilizations. From Fig. 19 we can see that the bounds become higher with the higher target utilizations. This is because the target utilization determines the bandwidth available on the ring for node 15. The higher the target utilization, the smaller the bandwidth “hole” left for node 15, and the larger the bound for the ring access delay.

Fig. 19 Ring access delay bounds with different target utilizations

After comparing the RPR bounds with the simulated ring access delay under concurrent greedy sessions in the four cases described above, we can see that all the simulated max ring access delays are well within their corresponding bounds. In comparison the worst case results as shown in Fig. 7 are very close to their corresponding bounds.

5.3 Simulation results for performance analysis of 1TB-RPR

The objective of the simulation is to examine the performance of 1TB-RPR as measured by ring access delay, utilization, and fairness. To achieve this objective, we investigated 1TB-RPR in mainly two kinds of applications, hub case application and distributed case application. In the hub case application, we studied ring access delay and utilization in both the transient status and static status. We also studied the fairness regarding the weight assigned to each node on the ring in the hub case application. In the distributed case application, we focused on the fairness issue as mentioned in Chapter 3 that the local traffic within a congestion span should not be affected by the advertised rate.

5.3.1 Hub application

In order to study ring access delay and utilization, we designed a worst-case scenario to test the algorithm. A hub application where all the nodes on the ring send traffic to one hub node is clearly more stressful than distributed applications. Furthermore we loaded the ring in such a way that the traffic load of each node jumps at the same time so that the target load of the last link raises from 50% to 100% instantaneously. We refer to this case as the Step Response scenario. The ring model is set up as shown in Fig. 20 where node 0 is the hub, and nodes 1 to 15 send traffic to node 0 along counter clockwise direction (inner ring).

The traffic model uses three traffic classes: Class A (also named EF), Class B (also named AF), Class C (also named BE). The packet inter-arrival distribution of AF and BE is exponential (Poisson traffic) while the packet inter-arrival distribution of EF is constant. While 444.4 bytes is the mean packet size, the packet size distribution for all classes is tri-modal, with sixty percent being 64 bytes, twenty percent 512 bytes, and twenty percent 1518 bytes.

This tri-modal distribution is believed to have captured the characteristics of Internet traffic [41].

Table 1 Common parameters set in the hub case

Target Utilization	95%
Ring Access Delay Threshold	1,000 us
Link Rate	10 G bps
Propagation Delay of one link	70 us

Fig. 20 Ring model for hub application

Fig. 21 Step response results

The common parameters for the simulation setup are described in Table 1. We ran the Step Response scenario for 0.5 simulated second. Initially, all nodes are in Normal state, and BE traffic from each node is only limited by the node itself. When the total traffic load jumps from 50% to 100% at 0.1 simulated seconds, node1 detects congestion and advertises a fair rate to its upstream nodes to control BE traffic. From the link utilization diagram shown in Fig. 21 and Fig. 22 (Large view of the link utilization), we note that 1TB-RPR can achieve 95% utilization and still work stably under 100% traffic load.

The ring access delay diagram indicates that in the transient period when the traffic load suddenly jumps from 50% to 100%, the ring access delays of several packets can range around 200 us to 600 us while all other packets suffer less than 200 us access delays. In most cases, it should be noted that this is still reasonably small and happens rarely during the transient period. In reality, this scenario cannot happen because real data traffic flows can hardly be synchronized in such a way that the total load would spike from 50% to 100% instantaneously.

Fig. 22 Utilization of the link 1 to 0 in the step response scenario

Fig. 23 Piecewise linear results

Under realistic traffic scenarios the ring access delay can be significantly lower if the mean traffic load changes slowly. To test this conclusion, we designed a so-called Piecewise Linear scenario in which the traffic load changes from 50% to 100% gradually within 150 ms. From Fig. 23 we can see that the maximum ring access delay of all the packets in the transient period of a Piecewise Linear scenario is three times lower than that in Step Response scenario. Fig. 24 shows a large view of the utilization of the link between node0 and node1 to indicate that 95% utilization is also achieved in this Piecewise Linear scenario.

Fig. 24 Utilization of the link 1 to 0 in the Piecewise linear scenario

Fig. 25 shows the Cumulative Distribution Function (CDF) of the EF packet ring access delay of Node1 at 95% target utilization and 100% traffic load. The CDFs indicate that 98% of all ring access delays are lower than 100 μ s with both 2000 bit and 150000 bits leaky bucket, and also shows that a smaller leaky bucket size can improve the performance in terms of ring access delay.

We now move on to study the ring access delay characteristic. Given the statistical nature of ring access delays, we study this performance metric under steady state conditions. Considering that ring access delay is affected by leaky bucket size of the nodes on the ring, we ran this hub case simulation in two scenarios with different leaky bucket sizes. In one scenario we set 2000 bits as the leaky bucket size to all the nodes on the ring, while setting 150,000 bits as the leaky bucket size for all the nodes in the other scenario.

Fig. 25 CDF of EF ring access delay of Node 1 at 95% target utilization and 100% load

From Fig. 25 we can see that, though the main parts of the total ring access delays are small in both scenarios, the max ring access delay in the scenario with 2000 bits leaky bucket size is much smaller than that in the scenario with 150,000 bits. Does that mean that the

smaller the leaky bucket size, the better the performance of 1TB-RPR? The answer is complicated because a smaller leaky bucket results in shorter bursts that a node can add on to the ring at one time. If the leaky bucket size is set too small, it can affect fairness on the ring. We use an example to explain this point. Assume that there are 15 nodes on the ring sending traffic to one hub node. If the traffic is BE traffic and each node has the same weight, each node should get equal bandwidth. Because the traffic is bursty, each node can not access the ring right away when local packets come. When blocked by the burst on the ring, the node claims its bandwidth by collecting tokens in its leaky bucket. When there is a bandwidth “hole” available, the node can burst as much as the tokens collected in the leaky bucket, and therefore get the provisioned bandwidth. If a node’s leaky bucket size is set too small, when the node is blocked by the burst on the ring, it may not collect enough tokens to claim the bandwidth because the leaky bucket is full. Therefore the node cannot burst enough traffic when the ring is idle. This leads to unfairness to this node. The unfairness can be expressed in this way, with a small leaky bucket size, the traffic throughput that the node adds on to the ring is smaller than the ring bandwidth available for this node. To prove that leaky bucket size can affect fairness, we set up two hub case scenarios with single class traffic (BE). The leaky bucket size of all nodes is set to 2000 bits in one scenario and 150,000 bits in the other scenario. The traffic load in both scenarios is 100% utilization, while the target utilization is 100%. In order to make 100% utilization on the link between node0 and node1, the local arrival traffic to each node is set to 0.667 Gbps considering the link capacity is 10 Gbps.

Fig. 26 Average throughputs of all the nodes on the ring with 2000 bits bucket size

Fig. 27 Average throughputs of all the nodes on the ring with 150k bits bucket size

Fig. 26 shows the throughputs of all the nodes with 2000 bits leaky bucket size. The throughput of node1, which is the nearest node to the hub (node0), is 0.1 Gbps lower than that of other nodes. Fig. 27 shows the throughputs of all nodes in the scenario with 150,000 bits leaky bucket size, which are quit flat compared to those with 2000 bits leaky bucket size.

Fig. 28 1TB-RPR support for unequal bandwidth requirements

Since the 1TB-RPR supports weighted bandwidth fairness, we now examine this feature in our simulations. The scenarios here are also based on the hub case application as described above, but have different weight distribution patterns among the nodes on the ring. It should be noted that the leaky bucket size in all nodes of all scenarios is set to 150,000 bits. In Fig. 28, diagrams A and C indicate the weight distribution pattern among the nodes on the ring, while diagrams B and D display the bandwidth each node gets when congestion happens. From the diagrams in Fig. 28 we can see that the pattern of throughputs for all nodes matches the pattern of their corresponding weights very well.

From the above we can also see that 1TB-RPR achieves excellent fairness among the nodes in the hub case application.

5.3.2 Distributed application

As mentioned in section 2, 1TB-RPR uses multiple Virtual Destination Queues (VDQs) to prevent a congestion span from affecting the local traffic within the span in distributed applications. Therefore when simulating distributed applications, we focused on comparing local traffic throughput within the congestion span of one BE transmit buffer with the throughput of two BE transmit buffers (one in-span and another out-of-span).

Fig. 29 shows the ring model we built for the distributed application having four hubs on the ring. In this model, all non-hub nodes send traffic along the counter clockwise direction (inner ring). Because congestion span only affects the local low priority traffic that can be controlled by the advertised fair rate, we used BE traffic in this distributed case simulation.

Fig. 29 Ring model for the distributed application

The traffic configuration is described in Table 2. The traffic generator of node1 (TG 1), as for all the nodes on the ring, sends 0.883 Gbps to Server1, 0.05 Gbps to Server2, 0.05 Gbps to Server3 and 0.05 Gbps to Server4, while the total traffic sent by one node is 1.033 Gbps. From Table 2 we can see that the traffic to Server1 is overloaded to 105.96% utilization, but Server 2, Server3 and Server4 are 60% loaded. That means only the link from node 1 to node 0 on the ring is congested. By our earlier definition, node1 is the head node. Because all the no-hub nodes from node1 to node15 send traffic to server1, the congestion span spans from node1 to node15. The local traffic flows within the span are:

- Traffic flows from node 13, 14, 15, to Server 4.

Table 2 Traffic configuration for the distributed application

Unit: Gbps

	Server 1	Server 2	Sever 3	Server 4	Total
TG 1	0.8830	0.0500	0.0500	0.0500	1.0330
TG 2	0.8830	0.0500	0.0500	0.0500	1.0330
TG 3	0.8830	0.0500	0.0500	0.0500	1.0330
TG 5	0.8830	0.0500	0.0500	0.0500	1.0330
TG 6	0.8830	0.0500	0.0500	0.0500	1.0330
TG 7	0.8830	0.0500	0.0500	0.0500	1.0330
TG 9	0.8830	0.0500	0.0500	0.0500	1.0330
TG 10	0.8830	0.0500	0.0500	0.0500	1.0330
TG 11	0.8830	0.0500	0.0500	0.0500	1.0330
TG 13	0.8830	0.0500	0.0500	0.0500	1.0330
TG 14	0.8830	0.0500	0.0500	0.0500	1.0330
TG 15	0.8830	0.0500	0.0500	0.0500	1.0330
Total	10.5960	0.6000	0.6000	0.6000	

Table 3 Theoretical throughputs for the distributed case

Unit: Gbps

	Server 1	Server 2	Server 3	Server 4	Total
TG 1	0.6764	0.0383	0.0383	0.0383	0.7913
TG 2	0.6764	0.0383	0.0383	0.0383	0.7913
TG 3	0.6764	0.0383	0.0383	0.0383	0.7913
TG 5	0.7117	0.0500	0.0403	0.0403	0.8423
TG 6	0.7117	0.0500	0.0403	0.0403	0.8423
TG 7	0.7117	0.0500	0.0403	0.0403	0.8423
TG 9	0.7497	0.0500	0.0500	0.0425	0.8921
TG 10	0.7497	0.0500	0.0500	0.0425	0.8921
TG 11	0.7497	0.0500	0.0500	0.0425	0.8921
TG 13	0.7921	0.0500	0.0500	0.0500	0.9421
TG 14	0.7921	0.0500	0.0500	0.0500	0.9421
TG 15	0.7921	0.0500	0.0500	0.0500	0.9421
Total	8.7894	0.5649	0.5358	0.5132	

- Traffic flows from node 9, 10, 11, 13, 14, 15, to Server 3.
- Traffic flows from node 5, 6, 7, 9, 10, 11, 13, 14, 15, to Server 2.

All those local traffic flows should not be affected by the congestion of node1. Table 3 shows what the theoretical throughputs of all the flows should be if the local traffic flows are not adversely affected by the congestion span. If the 1TB-RPR node is implemented with only one transmit queue for local BE traffic, all the BE traffic flows will be regulated by the advertised rate from the head node. Therefore the in-span local traffic flows will be reduced unnecessarily. The consequent result is that the drop-off throughputs of server 2, server 3 and server 4 will become lower than the ideal values when the local flows are not affected by the congestion span, as indicated in Table 3. This is clearly unfair to local traffic.

Fig. 30 demonstrates the significant improvement in local traffic throughput by using nodes having two transmit queues (one for in-span, one for out-of-span) for local BE traffic, versus the throughput of the single transmit queue scenario.

Fig. 30 Single queue vs. multiple (two) queues

Chapter 6 Conclusions and future work

This Chapter summarizes the conclusions and future work arising out of this thesis.

Resilient Packet Ring (RPR) is a new data transport technology to standardized as IEEE 802.17. There are several proposals made to the IEEE 802.17 Working Group on how the bandwidth and congestion management scheme should be implemented. The proposals differ in the number of transit buffers being implemented, and the way congestion is detected and controlled. We investigated the performance of one of the proposals, 1TB-RPR, which features single transit buffer, utilization-based congestion detection, and congestion notification with rates. The key benefit of 1TB-RPR is that it extremely simple to implement in hardware, since it require much smaller transit buffer size compared to other proposals. But on the other hand all add-in traffic streams' ring access delay will be affected under 1TB-RPR proposal, therefore the bound for ring access delay is a key issue of this proposal. In this thesis, we have mainly two conclusions. Firstly, from the research shown in chapter 4, we conclude that there is a bound for ring access delay of a 1TB-RPR system and the bound is related to the number of nodes on the ring, leaky bucket size of each node, token rate of each node, and target utilization. Secondly, the simulation results demonstrate that an RPR ring with single transit buffer can achieve more than 95% utilization with very low ring access delay. The 1TB-RPR-fairness scheme is stable and fair to all nodes under congestion. In addition, 1TB-RPR can support unequal bandwidth requirements and distributed applications effectively as predicted.

There are still many issues of 1TB-RPR to be studied as a future work.

First, future work should investigate more about the performance of the transient period of 1TB-RPR proposal. Secondly, it can be comparing the performance of Resilient Packet Rings with single transit buffer to those with multiple transit buffers. Thirdly, protection issues of 1TB-RPR are also good topics to be studied. Finally, future work should analyze the performance of 1TB-RPR with a multi-ring topology.

References

- [1] Cole, N., Hawkins, J., Green, M., Sharma, R. and Vasani, K., “Resilient Packet Rings for Metro Networks,” <http://www.rpralliance.org/>, August 2001.
- [2] RPR Alliance, “An Introduction to Resilient Packet Ring Technology,” <http://www.rpralliance.org/>, October 2001.
- [3] IEEE 802.17 Working Group, “Proposed IEEE Standard 802.17 – Darwin,” http://www.ieee802.org/17/documents/drafts/Darwin_v1_0.pdf, Draft version: January 2002.
- [4] IEEE 802.17 Working Group, “Proposed IEEE Standard 802.17 – Gandalf,” http://www.ieee802.org/17/documents/drafts/gandalf_04Plus.pdf, Draft version: November 2001.
- [5] IEEE 802.17 Working Group, “Proposed IEEE Standard 802.17 – Alladin,” <http://www.ieee802.org/17/documents/>, Draft version: November 2001.
- [6] Nortel Networks, “OPTera Packet Edge System General Specification,” Document release: 1.0, September 2000.
- [7] Knapp, E. M., “SONET Technology and Applications: a User Perspective,” *IEEE GLOBECOM 1989*, Dallas, Texas, USA, vol. 3, pp. 1488 – 1494, Nov.1989.
- [8] Ballart, R.and Ching, Y. C., “SONET: Now it is the Standard Optical Network,” *IEEE Communication Magazine*, vol. 27, pp. 8 – 15, Mar. 1989.
- [9] Maruta, R.; Hasegawa, S.; Higo, Y., “SONET Based Network Architecture and Its Applications,” *IEEE ICC 1990*, Atlanta, GA, USA, vol. 3, pp. 904 – 908, Apr. 1990.
- [10] Yan, S.; To, M.; Oxner, S., “Bandwidth Management in a SONET Transport Network,” *IEEE GLOBECOM 1990*, San Diego, USA, vol. 3, pp. 1257 – 1263, Dec. 1990.

- [11] To, M. and McEachern, J., "Planning and Deploying a SONET-Based Metro Network," *LTS 1991*, vol. 2, pp. 19 – 23, Nov. 1991.
- [12] Wu, T. H. and Lau, R. C., "A Class of Self-healing Ring Architectures for SONET Network Applications," *IEEE Transactions*, vol. 40, pp. 1746 – 1756, Nov. 1992.
- [13] Drake, T., "A Review of The Four Major SONET/SDH Rings," *IEEE ICC 1993*, Geneva, Switzerland, vol. 2, pp. 878 – 884, May 1993.
- [14] Tolly, B., "Moving the Decimal Point: 10 Gigabit Ethernet Applications and Market Opportunity," *Lasers and Electro-Optics Society 2000*, vol. 1, pp. 288 – 299, Feb. 2000.
- [15] Farrel, P. A. and Hong, O., "Communication Performance Over a gigabit Ethernet Network," *IPCCC 2000*, Phoenix, Arizona, USA, pp. 181 – 189, Feb. 2000.
- [16] William Stallings, "Data and Computer Communications," Fourth Edition, prentice Hall, 2000.
- [17] Varadarajan, S. and Chiuch, T., "Automatic Fault Detection and Recovery in Real Time Switched Ethernet Networks," *IEEE INFOCOM 1999*, New York, USA, vol. 1, pp. 161 – 169, Mar. 1999.
- [18] Kirstadter, A., "Fairness and Performance Limits of Contention Resolution Mechanisms for Input Buffered Switches," *ICCT 1998*, Beijing, China, vol. 1, pp. 64 – 66, Oct. 1998.
- [19] Vukovic, I. N. and Vastola, K. S., "An analysis of a New Carrier Senescing Scheme for Unidirectional Bus CSMA Networks," *IEEE ICC 1996*, Dallas, TX, vol. 2, pp. 1088 – 1092, June 1996.
- [20] Burr, W. E., "The FDDI Optical Data Link," *IEEE Commun. Mag.*, vol. 24, pp. 18 – 23, May 1986.
- [21] Ross, F. E., "FDDI- A Tutorial," *IEEE Commun. Mag.*, vol. 24, pp. 10 – 17, May 1986.

- [22] "Distributed Dual Bus (DQDB) Sub-network of a Metropolitan Area Network (MAN)," IEEE 802.6 Proposal and Submissions, Feb. 1990.
- [23] B. Mukherjee and C. Bisdikian, "A Journey through the DQDB Network Literature," *Performance Evaluation special issue on Performance Modeling of High-Speed Telecommunication Systems*, vol. 16, pp. 129-158, Dec. 1992.
- [24] Burr, W. E.; Wakid, S.; Qian, X. M., "A Comparison of FDDI Asynchronous Mode and DQDB Queue Arbitrated Mode Data Transmission for Metropolitan Area Network Applications," *IEEE Transactions*, vol. 42, Page(s): 1758 – 1768, Nov. 1994.
- [25] Chiu, A. L., and Gallager, R. G., "Full Utilization and Fairness on High Speed Bus Network," *IEEE GLOBECOM 1996*, London, England, vol. 1, pp. 508 – 512, Nov, 1996.
- [26] Cidon, I. And Ofek, Y., "MetaRing – A Full-Duplex ring with Fairness and Spatial Reuse," *IEEE/ACM Transaction on Communication*, vol. 41, pp. 110 – 120, Jan. 1993.
- [27] Ofek, Y.; Sohraby, K.; Wu, H., "Integration of Synchronous and Asynchronous Traffic on The MetaRing and Its Performance Study," *IEEE/ACM Transactions on Networking*, vol. 5, pp. 111 – 121, Feb. 1997.
- [28] Ofek, Y., "Overview of the MetaRing Architecture," *Computer Networks and ISDN System*, vol. 26, pp. 817 – 830, Mar. 1994.
- [29] <http://standards.ieee.org/catalog/IEEE802.5.html>.
- [30] Peng, H., "iPT Fairness: Control Access Protocol (CAP)," Northern Telecom Limited, Issue number: 1.0, May 2001.
- [31] Cruz, R. L., "A Calculus for Network Delay, Part I: Network Elements in Isolation," *IEEE Transaction, Inform. Theory*, vol. 37, pp. 114 – 131, 1991.

- [32] Cruz, R. L., “ A Calculus for Network Delay, Part II: Network Analysis,” *IEEE Transaction, Inform. Theory*, vol. 37, pp. 132 – 141, 1991.
- [33] Parekh, A. K. and Gallager, R. G., “A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case,” *IEEE/ACM Transaction on Networking*, vol. 1, iss. 3, pp. 344 – 357, June 1993.
- [34] Parekh, A. K. and Gallager, R. G., “A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Multiple-Node Case,” *IEEE/ACM Transaction on Networking*, vol. 2, iss. 2, pp. 137 – 150, Apr. 1994.
- [35] Yu, X.; Thng, I. L.; Jiang, Y.; Qiao, C., “Upper Bounds for Individual Queue Length Distribution in GPS with LRD Traffic Input,” *IEEE GLOBECOM 2001*, San Antonio, TX, USA, vol. 4, pp. 2336 – 2340, Nov. 2001.
- [36] Diwan, A.S.; Sivarajan, K.N., “Routing and Rate Allocation to Provide End-to-End Delay Guarantees under PGPS Scheduling,” *IEEE ICC 2001*, Helsinki, Finland, vol. 7, pp. 2128 – 2133, June, 2001.
- [37] Sivaraman, V.; Chiussi, F.M.; Gerla, M., “Traffic Shaping for End-to End Delay Guarantees with EDF Scheduling,” *IEEE/IFIP IWQOS 2000*, Pittsburgh, PA, pp. 10 – 18, June 2000.
- [38] Valaee, Shahrokh, “A recursive estimator of worst-case burstiness,” *IEEE/ACM Transactions on Networking*, vol. 9, iss. 2, pp. 211 – 222, Apr. 2001.
- [39] Andrews, M., “Probabilistic end-to-end delay bounds for earliest deadline first scheduling,” *IEEE INFOCOM 2000*, Tel Aviv, Israel, vol. 2, pp. 603 – 612, Mar. 2000.
- [40] Starobinski, D.; Sidi, M., “Stochastically bounded burstiness for communication networks,” *IEEE INFOCOM 1999*, New York, USA, , vol. 1, pp. 36 – 42, Mar. 1999.

- [41] Cidon, I., Georgiadis, L., Gurin, R., and Shavitt, Y., "Improved Fairness Algorithms for Rings with Spatial Reuse", *IEEE/ACM Transactions on Networking*, vol. 5, iss. 2, pp.190-204, Apr. 1997.
- [42] Kurose, J. F. and Ross, K. W. "Computer Networking: A Top-Down Approach Featuring the Internet," Pearson, 2001.
- [43] Karvelas, D., and Leon-Garcia, A., "Performance Analysis of the Medium Access Control Protocol of the FDDI Token Ring Network," *IEEE GLOBECOM 1988*, Hollywood, FL, vol. 2, pp. 1119 – 1123, Nov. 1988.