Scheduling Algorithms for Resilient Packet Ring Networks with Video Transport Applications

Jian Zhu

**M.Eng. (Electrical Engineering)
Shanghai University, China**

**B.Eng. (Electrical Engineering)
University of Electronic Science and Technology of China**

**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 for Electrical and Computer Engineering
Faculty of Engineering**

**Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, K1S 5B6, Canada**

December 2004

© Jian Zhu

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: 0-494-00756-7

Our file Notre référence

ISBN: 0-494-00756-7

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

The Resilient Packet Ring (RPR) architecture, known as the IEEE 802.17 standard, is an emerging network architecture and technology designed to meet the requirements of a packet-based metropolitan area network. The Priority Queue (PQ) algorithm is recommended as the scheduling scheme for RPR, which always gives priority to the transit buffer. Using this scheduling scheme in RPR, the high priority traffic, such as video packets waiting to access the ring at congested node in the transmit buffer, will suffer large delays and unsteady delay jitters. In this thesis, we propose new scheduling schemes which alternately select packets from the transit buffer and the high priority transmit buffer using the Deficit Round-Robin (DRR) algorithm which is known to be of low complexity. We investigate the system performance for the case of transmitting MPEG-4 encoded video as high priority traffic. Simulation results show certain improvements in the overall delay, delay jitter, and video quality.

ACKNOWLEDGEMENTS

With a deep sense of gratitude, I wish to express my sincere thanks to both of my supervisors, Professor Ioannis Lambadaris and Professor Ashraf Matrawy, for their guidance, encouragement, patience and continuous support. What I know today about the process of research, I learned from them. Without their valuable suggestions, I could not be productive in generating new ideas.

Special thanks to our graduate assistant Blazenka Power. Although it is not her responsibility, she proofread my thesis in her spare time.

I also want to thank Mohsen Ashourian and Arash Shokrani for their suggestions and help on MPEG-4 and RPR simulation. Also I am grateful to all other friends for their friendship and their support.

Last but not least, my deepest thanks go to my family, for their love and trust.

TABLE OF CONTENTS

1. INTRODUCTION.....	1
1.1 BACKGROUND AND MOTIVATION	1
1.2 OBJECTIVES OF THIS THESIS.....	2
1.3 THESIS CONTRIBUTIONS.....	3
1.4 THESIS OUTLINE	4
2. RESILIENT PACKET RING (RPR) NETWORKS	6
2.1 OVERVIEW OF RPR	7
2.2 THE SINGLE TRANSIT BUFFER (STB) ARCHITECTURE	11
2.3 THE DUAL TRANSIT BUFFER (DTB) ARCHITECTURE.....	13
2.4 KEY CHARACTERISTICS	14
3. SCHEDULING ALGORITHMS.....	17
3.1 PROCESSOR SHARING SCHEDULING ALGORITHMS.....	17
3.1.1 <i>Virtual Clock (VC)</i>	18
3.1.2 <i>Packet-by-Packet Generalized Processor Sharing (PGPS)</i>	19
3.1.3 <i>Self-Clocked Fair Queuing (SCFQ)</i>	20
3.2 ROUND-ROBIN SCHEDULING ALGORITHMS	20
3.2.1 <i>Round Robin (RR) Algorithm</i>	20
3.2.2 <i>Weighted Round Robin Algorithm (WRR)</i>	21
3.2.3 <i>Deficit Round Robin Algorithm (DRR)</i>	21
3.3 DISCUSSION	23

4. MODELING AND SIMULATION OF MPEG-4 TRAFFIC	25
4.1 OVERVIEW OF MPEG-4	26
4.2 TRANSFORM EXPAND SAMPLE (TES) METHODOLOGY	27
4.3 MODELING AND SIMULATION.....	30
4.3.1 <i>Variable Quantizer Rate Control</i>	32
4.3.2 <i>Variable Layer Rate Control</i>	34
5. NEW SCHEDULING SCHEMES FOR RESILIENT PACKET RING NETWORKS.....	36
5.1 THE PROPOSED SCHEDULING SCHEME FOR STB.....	37
5.1.1 <i>STB scheduling scheme</i>	38
5.1.2 <i>Simulations of the Proposed Scheduling Scheme for STB</i>	39
5.1.2.1 Hub Scenario.....	39
5.1.2.2 Mesh Scenario.....	43
5.2 THE PROPOSED SCHEDULING SCHEME FOR DTB.....	47
5.2.2 <i>DTB scheduling scheme</i>	48
5.2.2 <i>Simulation of the Proposed Scheduling Scheme for DTB</i>	49
6. EVALUATION OF MPEG-4 TRANSMISSION OVER THE NEW SCHEDULING SCHEME	52
6.1 DELAY AND DELAY JITTER EVALUATION	53
6.2 EVALUATION USING PSNR	64
7. CONCLUSIONS AND FUTURE RESEARCH.....	68
7.1 CONCLUSIONS	68

7.2	FUTURE RESEARCH.....	70
	BIBLIOGRAPHY	71

LIST OF FIGURES

<i>Figure 1. RPR Network Topology.....</i>	<i>8</i>
<i>Figure 2. The MAC architecture of RPR with Single Transit Buffer.....</i>	<i>12</i>
<i>Figure 3. The MAC Architecture of RPR with Dual Transit Buffer</i>	<i>13</i>
<i>Figure 4. I-frames Model.....</i>	<i>30</i>
<i>Figure 5. Comparison Between the Original and the Generated Traces</i>	<i>31</i>
<i>Figure 6.Original Traces of I Frames with Different Quantization Table.....</i>	<i>33</i>
<i>Figure 7.Comparison between Original Trace of I Frame (Q=10) and Trace Modeled from Q=4.</i>	<i>34</i>
<i>Figure 8.The Single Transit Buffer MAC Architecture of the Proposed Scheme.</i>	<i>38</i>
<i>Figure 9. The Delay of HP Traffic using the Original RPR Scheduling Scheme (Hub Scenario).....</i>	<i>41</i>
<i>Figure 10. The Delay of HP Traffic Using the Proposed RPR Scheduling Scheme (Hub Scenario).....</i>	<i>41</i>
<i>Figure 11. The Delay Jitter of HP Traffic Using the Original RPR Scheduling (Hub Scenario).....</i>	<i>42</i>
<i>Figure 12. The Delay Jitter of HP Traffic Using the Proposed RPR Scheduling Scheme (Hub Scenario).....</i>	<i>42</i>
<i>Figure 13 The Delay of HP Traffic Using the Original RPR Scheduling Scheme (Mesh Scenario).....</i>	<i>45</i>
<i>Figure 14 The Delay of HP Traffic Using the Proposed RPR Scheduling Scheme (Mesh Scenario).....</i>	<i>45</i>

<i>Figure 15 The Delay Jitter of HP Traffic Using the Original RPR Scheduling (Mesh Scenario).....</i>	<i>46</i>
<i>Figure 16 The Delay Jitter of HP Traffic Using the Proposed RPR Scheduling Scheme (Mesh Scenario).....</i>	<i>46</i>
<i>Figure 17 The Dual Transit Buffer MAC Architecture of the Proposed Scheme.</i>	<i>49</i>
<i>Figure 18 Comparison of Congested Node (node 9) High Priority Transmit Buffer Throughput by Using Different Scheduling Schemes</i>	<i>51</i>
<i>Figure 19. Comparison of End-to End Delay of I frames of Video “News” by Using Different Scheduling Schemes.....</i>	<i>54</i>
<i>Figure 20. Comparison of End-to End Delay of P frames of Video “News” by Using Different Scheduling Schemes.....</i>	<i>54</i>
<i>Figure21. Comparison of End-to End Delay of B frames of Video “News” by Using Different Scheduling Schemes.....</i>	<i>55</i>
<i>Figure 22. Comparison of End-to End Delay of I frames of Video “Akiyo” by Using Different Scheduling Schemes.....</i>	<i>56</i>
<i>Figure 23. Comparison of End-to End Delay of P frames of Video “Akiyo” by Using Different Scheduling Schemes.....</i>	<i>56</i>
<i>Figure 24. Comparison of End-to End Delay of B frames of Video “Akiyo” by Using Different Scheduling Schemes.....</i>	<i>57</i>
<i>Figure 25 Comparison of Delay Jitter of Video “News” by Using Different Scheduling Schemes.....</i>	<i>59</i>
<i>Figure 26. Comparison of Delay Jitter of Video “Akiyo” by Using Different Scheduling Schemes.....</i>	<i>60</i>

<i>Figure 27 Comparison of Occupied High Priority Transmit Buffer Size of Node 9 (Congested Node) in Two Schemes.....</i>	<i>61</i>
<i>Figure 28 Comparison of Occupied Transit Buffer Size of Node 9 (Congested Node) in Two Schemes.....</i>	<i>62</i>
<i>Figure 29 Comparison End to End Delay of High Priority Traffic from all Nodes.....</i>	<i>63</i>
<i>Figure 30 Comparison of PSNR of Video “Akiyo” by Using Different Scheduling Schemes.....</i>	<i>66</i>
<i>Figure 31 Comparison of PSNR of Video “News” by Using Different Scheduling Schemes.....</i>	<i>67</i>

Abbreviation

DRR	Deficit Round Robin
DTB	Dual Transit Buffer
FFQ	Fluid Fair Queuing
HOL-PS	Head of Line Processor Sharing
HP	High Priority
LAN	Local Area Networks
LP	Low Priority
MAC	Medium Access Control
MAN	Metropolitan Area Networks
PGPS	Packet-by-Packet Generalized Processor Sharing
QoS	Quality of Service
RPR	Resilient Packet Ring
RR	Round Robin
PSNR	Peak Signal to Noise Ratio
SCFQ	Self Clocked Fair Queuing
STB	Single Transit Buffer
TES	Transform Expand Sample methodology
TDM	Time Division Multiplexing
TTL	Time To Live
VC	Virtual Clock
WRR	Weighted Round Robin

Chapter 1

1. Introduction

1.1 Background and Motivation

The Resilient Packet Ring (RPR) [1][2] architecture (IEEE 802.17 standard), is a ring network technique based on a highly scaleable and resilient technology for the efficient transfer of packet-based traffic across a network. RPR networks offer multiple performance benefits such as spatial reuse, bandwidth efficiency, ease of management, resilience and scalability. These advantage make RPR an attractive alternative for high speed MANs (Metropolitan Area Networks) and LANs (Local Area Networks).

RPR has two ring architectures: Single Transit Buffer (STB), and Dual Transit Buffer (DTB) [1][2][3][4]. For both architectures, a high priority and a low priority transmit buffer exist at the client side. In the STB architecture, a single transit buffer mixes Low Priority (LP) and High Priority (HP) traffic from the transmit buffer into one traffic buffer on the ring. In the DTB architecture, two transit buffers exist for LP and HP traffic. When the traffic arrives from the ring, it is either dropped if it is destined for that node, or placed in the high priority or low priority transit buffer, according to its classification.

The Priority Queue (PQ) [5] algorithm is used as the scheduling scheme for RPR. The scheduler serves a specific queue on the basis that all the higher priority queues have been handled and emptied. This approach ensures low delay for higher priority traffic. However, it often causes the starvation of lower priority queues [6]. For the STB architecture, traffic on the ring, which is a mixture of HP and LP transit traffic, has higher priority over the transmit HP traffic. This could cause the LP traffic on the ring to block the transmit HP traffic from accessing the ring. For the DTB architecture, overwhelming HP traffic from upstream node, which is not subject to the fairness algorithm, can block downstream HP traffic from gaining access onto the ring. For both architectures, high priority traffic waiting to access the ring at the congested node will suffer large delays and unsteady delay jitters. Video traffic, usually classified as high priority traffic in RPR, is delay and delay jitter sensitive. Therefore, RPR traditional scheduling schemes may not satisfy the strict Quality of Service (QoS) requirements of real-time video applications.

1.2 Objectives of this Thesis

- Investigation of scheduling algorithms to improve the performance of the high priority traffic in RPR: Video traffic classified as high priority traffic in RPR has very high QoS requirement. Current scheduling algorithm in RPR is not fair for the high priority traffic during network congestion interval. To improve the QoS of video traffic transmission in RPR, we need to investigate all the popular scheduling algorithms and propose one which can improve the performance of high priority traffic in RPR.
- Modeling and Simulation of MPEG-4 traffic: MPEG-4 traffic is needed as traffic sources for network performance study. A traffic generation tool which can

model and simulate real MPEG-4 traffic is an ideal substitute for real MPEG-4 traces.

- Studying performance impact of scheduling algorithms on MPEG-4 transmission:

To prove the effectiveness of our proposed scheduling algorithm, we need to evaluate the quality of MPEG-4 traffic after transmission over different scheduling algorithms. Delay, delay jitter and objective measure Peak Signal to Noise Ratio (PSNR) are used to measure the performance impact of scheduling algorithms.

1.3 Thesis Contributions

The following are the main contributions of this thesis:

- New scheduling schemes for RPR based on the Deficit Round Robin algorithm [7]. Two scheduling schemes are proposed separately for RPR with single transit buffer and dual transit buffer. Our scheme can provide guaranteed service for high priority traffic at congestion nodes. [8].
- Performance evaluation of our proposed schemes through simulations. The performance was compared with the original RPR priority queue scheduling schemes. Through the simulation results, we show that our schemes achieve improvement over the original RPR schemes on delay and delay jitter performance for video traffic at congested nodes [9].
- Modeling traces of real MPEG4 traffic. The models are developed by using Transform Expand Sample (TES) [10]. We model I frames, P frames and B frames using three different TES models as in [11]. Based on these models, we develop tools

and incorporate these tools into network simulator OPNET for MPEG-4 traffic generation. This enables the study of MPEG-4 transmission performance over networks by using simulation. The traffic we generate matches the statistical characteristics (in terms of marginal distribution and auto-correlation function) of an original real trace of video frames that were generated using an MPEG-4 encoder [12].

- Evaluation of the video quality after its transmission over networks using different scheduling schemes. Peak Signal to Noise Ratio (PSNR) is used as an objective measure of video quality. We observe better PSNR average and variance by using our proposed scheme due to less delay and jitter.

1.4 Thesis Outline

The rest of the thesis is organized as follows:

Chapter 2 introduces the RPR with single transit buffer and dual transit buffer. We also introduce the fairness algorithm, topology discovery, and resilience in RPR.

Chapter 3 presents an overview of scheduling algorithms. Processor sharing scheduling algorithm and round robin scheduling algorithm are introduced separately. We compare these two classes of scheduling algorithm at the end of this chapter.

Chapter 4 introduces our models and tools for simulation of video transmission. Transform Expand Sample methodology is used to model the real video traffic. A traffic generation tool is developed based on those models. We incorporate the traffic generation tool into OPNET simulation platform as traffic source.

Chapter 5 explains our proposed scheduling schemes. We introduce two scheduling schemes for single transit buffer and dual transit buffer separately. Simulation results and analysis are presented for both the proposed scheduling schemes.

Chapter 6 presents simulation results and analysis on video transmission on RPR. We compare the results of our proposed scheduling scheme with the original RPR priority queue scheduling schemes.

Chapter 7 concludes the thesis.

Chapter 2

2. Resilient Packet Ring (RPR) Networks

The Resilient Packet Ring (RPR) architecture, which is being standardized as IEEE 802.17, is a ring based network protocol supporting data transfer among nodes interconnected in a dual ring configuration. The RPR protocol scales to the demands of the future packet networks and includes sophisticated resilience mechanisms and efficient transfer of packet-based traffic across a network. RPR supports up to 256 node attachments and is optimized for rings with a maximum circumference of 2000 kilometers [1][2].

RPR has certain advantages over other MAN or WAN technologies such as SONET/SDH rings and Gigabit Ethernet. Most of the networks deployed by service providers as MANs or WANs are based on SONET/SDH rings. One of the rings in SONET is used as the back-up ring which remains unused during normal operation and utilized only in the case of failure of the primary ring. Network monitoring requirements and static bandwidth allocation increase the total cost of a SONET network. Gigabit Ethernet does not require static allocation and provides cost advantages, but it cannot provide the desired features such as fairness and auto-restoration [13].

2.1 Overview of RPR

RPR connects nodes into a point-to-point, bi-directional full duplex ring topology. Bidirectional rings provide resilience since a packet can reach its destination even in a situation of link failure in this kind of architecture. Unlike the token ring, in RPR several packets might be on their way at the same time, provided the packets use different segments of the ring. This is called spatial reuse [14]. The overall throughput of a network using spatial reuse can be significantly higher than that of a network without spatial reuse. RPR spatially reuse the ring bandwidth by letting the destinations strip the packets. If the destination address in the packet header does not match a node's address, the packet is forwarded to the next downstream node on the ring. Hence one packet may flow on one segment of a ringlet while other packet flows on another part of the same ringlet at the same time. Here a ringlet is defined as a subnet that connects all the nodes and moves traffic in one direction around the ring [15]. Each node is connected to two ringlets, and has a full duplex connection to the outside. The clockwise ringlet is denoted as the O-ring (outer ring) whereas the counter clockwise ring is denoted as the I-ring (inner ring). A typical RPR network topology is showed in Fig. 1.

Each ringlet interface includes transmit, transit and drop buffers. Packets on their way into the ring are stored in the transmit buffer, while packets that are stripped from the ring are stored in the drop buffer. Packets that are traveling on the ring are stored in each node's transit buffer to be processed.

Figure 1. RPR Network Topology

RPR supports two transit methods: cut-through (the node starts to forward the packet before it is completely received) and store-and-forward. When an RPR node is the receiver of a packet, it removes the packet completely from the ring, instead of merely copying the contents of the packet and letting the packet traverse the ring back to the sender as in the token ring. To prevent packets without a destination address recognized by any node on the ring from circulating forever, a Time to Live (TTL) field is decremented by all nodes on the ring [13].

The potential application of RPR will rely on its ability to provide QoS guarantees. These guarantees usually are bound on end-to-end delay, bandwidth, delay jitter (delay variation), packet loss rate, or a combination of these parameters. Providing QoS guarantees in a packet network requires the use of scheduling algorithms. The function of

a scheduling algorithm is to select the packet to be transmitted in the next cycle from the available packets belonging to the flows sharing the output link. We will introduce scheduling algorithm in details according to different ring architectures in the following two sections.

Access to the ring is controlled by a fairness or Media Access Control (MAC) algorithm. In RPR, a control system with explicit rate feedback to upstream nodes contributing to the congestion is used in order to avoid starvation and achieve fair access to the ring. The basis of such a control mechanism is flow control packets that a node sends upstream when it does not get its fair share of the bandwidth. When defining fair distribution of bandwidth, RPR uses a principle in which when the demand for bandwidth on a link is greater than the capacity, the available bandwidth should be fairly distributed between the contending sender nodes. A weight is assigned to each node so that a fair distribution of bandwidth does not need to be an equal one. Fair rate is calculated by dividing the available bandwidth between all upstream nodes that are currently sending packets through the congested node, or by using their own current add rate of the congested node. When the upstream nodes receive flow control packets, they adjust leak rates of their leaky bucket to the fair rate advertised in the flow control packets. It is important to note that only low priority traffic is subjected to the fairness algorithm and RPR does not discard packets to resolve congestion [13][15].

For fairness algorithm, there are two options [1][2][13]. In the “Conservative” mode, the congested node only sends out a new fair rate value until all nodes in the congestion node have adjusted to the fair rate, and this change is observed by the congested node itself. The round trip time needed for fair rate packet traveling across all

the nodes is estimated by sending special control frames across the congestion domain. In the “Aggressive” mode, fair rate calculations are made every sample period, which is typically set to 100 μ s. When a node receives a fairness flow packet indicating no congestion, it will gradually increase its send rate again. The conservative method adapts to the new traffic pattern with fewer oscillations but more slowly than the aggressive mode. In our simulation, we chose the aggressive fairness algorithm.

There are topology databases in each node of RPR [13]. Topology discovery protocol is used to collect information about the nodes and interconnecting links. All nodes send control packets at system initialization containing their own status around the ring. All nodes will be able to compute a complete topology image after receiving the control packets. Whenever a new node is inserted into a ring, or whenever a node detects a link failure, it will immediately transmit a topology discovery message. If any node receives a message inconsistent with its current topology image, it also will transmit a new topology message. This will result in all nodes transmitting their updated status information and all nodes rebuilding their topology image. The MAC also uses the topology database to find the shortest path when it needs to specify which ringlet to use for a packet.

RPR uses a behavior called steering to provide resilience. When a node detects that one of its links or a neighbor node has failed, it sends out topology messages. A node received such a message indicating that the ring is broken will start to send packets in the only viable direction to the receiver [13].

Each node of RPR has a high priority and low priority transmit buffer at the client side. Real time data such as video and voice can be classified as high priority traffic,

which is sensitive to delay and delay jitter. Low priority traffic can be attributed to other non-critical data, which may suffer from delay and packet loss, but still meets the application requirements. The transit buffers can be configured in such a way that it can have one or two transit buffers.

2.2 The Single Transit Buffer (STB) Architecture

In the single transit buffer architecture, a transit buffer combines mixed low priority and high priority traffic. The traffic in this buffer has the highest priority. Fig. 2 shows the single transit buffer MAC architecture. In Fig. 2 only one ringlet is shown, but the architecture of the other ringlet is similar. At each node, both ringlets share the same transmit buffers and the drop buffer. The advantage of this configuration is that it simplifies the hardware implementation since only one transit buffer is used and fewer parameters for fairness control are needed. However, the traffic on the ring, which is a mixture of high priority and low priority traffic, can block the transmitted high priority and low priority traffic waiting to access the ring.

In each RPR node, the scheduler selects traffic in the following order [4]:

- 1) Packets from the mixed transit buffer.
- 2) Packets from the high priority transmit buffer.
- 3) Packets from the low priority transmit buffer.

In the STB architecture, congestion can be triggered in two ways [1][2][4]:

- 1) If the link traffic exiting a node scheduler exceeds a threshold.

The threshold is typically set to 95% of the link's bandwidth. The congestion status is lifted when the link usage falls below the threshold.

Figure 2. The MAC architecture of RPR with Single Transit Buffer

2) If the head of line timer expires.

The head of line timer indicates whether a packet at the head of the low priority transmit buffer has waited a fixed length of time and is deemed unfair. The node is deemed congested when the head of line timer expires. When congestion occurs, the initial fair rate is calculated by

$$\rho_i = \frac{CU_t}{\omega} \quad (2.1)$$

ρ_i is the fair rate at node i , where node i is the congested node. C is the link rate of the ring. U_t is the link utilization. ω is the number of nodes contributing to the congestion. The congested node is required to keep track of the contributing nodes counted in ω .

Source nodes which generate traffic passing through the congested node are counted. The count is reset every sample period.

2.3 The Dual Transit Buffer (DTB) Architecture

In the DTB architecture, two transit buffers exist. One buffer is for low priority while the other is for high priority traffic. Fig. 3 shows the DTB MAC architecture. The arriving ring traffic is either dropped if destined for that node, or placed in the HP or LP transit buffer, according to its traffic classification. In dual transit buffer architecture, high priority traffic has the highest priority and is never blocked by low priority traffic passing through the node.

Figure 3. The MAC Architecture of RPR with Dual Transit Buffer

DTB MAC architecture requires more complicated hardware design and more parameters for fairness control compared to STB MAC architecture. Also the LP transit

buffer must be large enough to allow traffic bursts from the other buffers from contributing nodes.

The scheduling algorithm in DTB scheme is as follows [4]:

- 1) Packets from the High Priority Transit Buffer.
- 2) Packets from the High Priority Transmit Buffer.
- 3) Packets from the Low Priority Transmit Buffer.
- 4) Packets from the Low Priority Transit Buffer.

In DTB Scheme, congestion is triggered if the LP transit buffer size exceeds a threshold. There are two thresholds defined: The LO_THRESHOLD and the HI_THRESHOLD. When the LP transit buffer size exceeds the LO_THRESHOLD, then the node is deemed congested. If the buffer size exceeds HI_THRESHOLD, then the LP transit traffic is deemed starved and transmitted LP traffic is blocked to let the LP transit buffer access the ring. If, at a later time, the buffer size falls below LO_THRESHOLD, congestion is lifted. DTB scheme can achieve 100% link utilization because passing through LP traffic can always be buffered on the ring.

The fair rate is calculated by monitoring the add-in traffic from the congested node during congestion. This add-in rate is set as the fair rate in the fairness message. The basic logic of this algorithm is that all nodes should have the same transmitted throughput. If the throughput of a congested node drops, all other nodes should follow.

2.4 Key Characteristics

RPR has the following key features that can distinguish it from other network interconnects [2]:

- a) Addressing: Unicast, multicast, and simple broadcast data transfers are supported.

b) Services: Multiple service qualities are supported.

1) High Priority: The allocated/guaranteed bandwidth has low circumference-independent jitter.

2) Low Priority: Provides best-effort services.

c) Efficiency: Design strategies increase effective bandwidths beyond those of a broadcast ring.

1) Concurrent transmission: Clockwise and counter-clockwise transmissions can be concurrent.

2) Bandwidth reallocation: Bandwidths can be reallocated on non-overlapping segments.

3) Bandwidth reclamation: Unused bandwidths can be reclaimed by opportunistic services.

4) Spatial bandwidth reuse: Opportunistic bandwidths are reused on non-overlapping segments.

5) Temporal bandwidth reuse: Unused opportunistic bandwidth can be consumed by others.

d) Fairness: Fairness ensures proper partitioning of opportunistic traffic.

1) Weighted: Weighted fair access to available ring capacity.

2) Simple: Point-of-congestion flow-control facilitates per destination queuing in client.

3) Detailed: Allows the client to selectively throttle its transmissions based on multiple congestion point indications.

e) Plug-and-play: Automatic topology discovery and advertisement of station capabilities allow systems to become operational without manual intervention.

f) Robustness: Multiple features support robust frame transmissions.

1) Responsive: Service restoration time is less than 50 milliseconds after a station or link failure.

2) Lossless: Queue and shaper specifications avoid frame loss in normal operation.

3) Tolerant: Fully distributed control architecture eliminates single points of failure.

4) OAM: Operations, administration, and maintenance support service provider environments.

Chapter 3

3. Scheduling Algorithms

Scheduling algorithms are techniques that allow each flow passing through a network link to have a fair share of network resources. Based on its implementation strategies, most of the proposed work-conserving scheduling algorithms can be classified into two categories [16]: 1) schedulers that implement different variations of processor sharing service discipline [17] and 2) round-robin schedulers. In schedulers implementing different variations and approximations of processor sharing service discipline, Virtual Clock (VC) [18] , Packet-by-Packet Generalized Processor Sharing (PGPS) or Weighted Fair Queuing (WFQ) [19] and Self-Clocked Fair Queuing (SCFQ) [20] have received the most attention. Weighted Round Robin (WRR) [21] and Deficit Round Robin (DRR) [7] fall into the class of round-robin schedulers.

3.1 Processor Sharing Scheduling algorithms

Processor sharing scheduling algorithms assign time stamps to each packet arriving to the system. The time stamp is used by the scheduler to determine the sequence in which packets are served. VC, PGPS and SCFQ belong to this category.

3.1.1 Virtual Clock (VC)

The Virtual Clock scheduling algorithm [18] [22] is aimed to integrate the benefits of time-division multiplexing (TDM) with those of statistical multiplexing. In VC implementation, each packet is allocated a virtual transmission time, which is the time at which the packet would have been transmitted if the server were actually doing TDM. The service discipline of the Virtual Clock scheduling algorithm is that whenever the scheduler is ready to serve a new packet, the packet with the smallest virtual clock value is selected for service. This service discipline is work-conserving and non-preemptive.

The virtual clock of a packet in flow f is denoted by $P_m(f)$ [18][23].

$$P_m(f) := \max(P_{m-1}(f), A_m(f)) + l_m(f)/r(f) \quad (3.1)$$

Here m is an arbitrary packet of flow f . We use $l_m(f)$ to denote its packet length in bits, and $A_m(f)$ its arrival time to the facility ($A(f) \geq 0$). Flow rate $r(f)$ bits/second is generated by negotiation between the source of flow f and the service facility based on needs and how much they are willing to pay. $P_m(f)$ is implemented as a variable, which is zero initially and updated whenever a flow f packet, such as m , arrives to the facility. The virtual clock of flow f is a variable that holds the virtual clock value of the most recent arrival of f . For a flow with traffic at exactly the reserved rate, Virtual Clock will equal the real time. In implementation, each guaranteed flow is allocated its own buffers, in order that the delay guarantee to a flow is independent of the behavior of other flows that share the same service facility. This is called Firewall Property.

3.1.2 Packet-by-Packet Generalized Processor Sharing (PGPS)

PGPS [19] is a scheduling scheme which approximates Fluid Fair Queuing (FFQ) [17] service in a packet system. FFQ is a general form of the head-of-line processor sharing service discipline (HOL-PS) [22][24]. In HOL-PS, there is a separate FIFO queue for each flow sharing the same link. During any time interval, when there are N non-empty flows, the server serves the N packets at the head of the flows simultaneously, each at a rate of one N th of the link speed. While a HOL-PS server serves all non-empty flows at the same rate, FFQ allows different connections to have different service shares in proportion to their service shares. The FFQ scheduler visits each active flow's queue in a round-robin fashion and serves an infinitely small amount of data from each queue in such a way that during any finite interval of time, it can visit each queue at least once[25]. Consider a set of N flows denoted by $1, 2, \dots, n$ demanding bandwidths $b_1, b_2, \dots, b_n$ on a link of total bandwidth B . Without loss of generality, assume $b_1 \leq b_2 \leq \dots \leq b_n$. The FFQ scheduler first allocates B/n of the bandwidth to each of the active flows. If this is more than the bandwidth demanded by flow 1, the unused bandwidth, $B/n - b_1$, is divided equally among the remaining $n-1$ flows. The allocation process of the FFQ scheduler continues in this fashion until each flow has received no more than its demand and, if the demand was not satisfied, no less than any other flow with higher demand. FFQ is impractical because it assumes that the server can serve all connections with nonempty queues simultaneously and that the traffic is indefinitely divisible. In a more realistic packet system, only one flow can receive service at a time and an entire packet must be served before another packet can be served.

PGPS approximate FFQ service in the following way [22]: when the server is ready to transmit the next packet at time τ , it picks, among all the packets queued in the system at τ , the first packet that would complete service in the corresponding FFQ system if no additional packets were to arrive after time τ .

3.1.3 Self-Clocked Fair Queuing (SCFQ)

PGPS need to emulate a reference FFQ server. However, maintaining the reference FFQ server is computationally expensive. Self-Clocked Fair Queuing (SCFQ) [20][22] is one simpler packet approximation algorithm of FFQ.

SCFQ introduces a Virtual time approximation algorithm. The algorithm is based on the observation that the system's virtual time at any moment may be estimated from the virtual service time of the packet currently being serviced. SCFQ is slightly less fair than PGPS and also has a larger delay bound [20].

3.2 Round-robin Scheduling Algorithms

Round-robin scheduler serves flows in rounds. An active flow can send out one packet during each round. A flow is deemed active if there is at least one packet in its buffer waiting to be served. RR, WRR and DRR belong to this category.

3.2.1 Round Robin (RR) Algorithm

Round Robin algorithm [26] is the base of all other scheduling algorithms which serve flows in rounds. It circularly scans all the flow and transmits one packet from each active flow in each round. RR algorithm distributes all the available link capacity equally to all the active flows. It is fair insofar as all contending flows have the same reserved

rate and all packets have the same size. For flows with variable packet sizes and different reserved rates, RR algorithm is not appropriate.

3.2.2 Weighted Round Robin Algorithm (WRR)

Weighted round robin (WRR) [21] scheme applies a weight for each flow. During each cycle, the scheduler visits all flows in a predetermined order. When the scheduler starts to process a flow, it de-queues a number of packets according to the corresponding weight of this flow before proceeding to the next flow. WRR doesn't consider the variation of packet lengths [21]. It assumes that the average packet size over the duration of a cycle is the same for all flows. If this is true, each flow gets an equal share of the output link bandwidth. However, a flow can get *max/min* times the bandwidth of another flow in the worst case. Here *max* is the maximum packet size of one flow and *min* is the minimum packet size of another flow. To cope with this problem, Deficit Round Robin (DRR) algorithm is proposed.

3.2.3 Deficit Round Robin Algorithm (DRR)

The DRR [7] algorithm is an extension of the WRR algorithm. Instead of processing a certain number of packets from each flow, DRR assigns a quantum for each flow. The quantum allocated to a flow is defined as the service that the flow should receive during each round robin service opportunity.

Active List, which is a linked list of the active flows, is maintained in the DRR scheduler [27]. When a flow is activated by receiving one or more packets, the flow is added to the tail of the Active List. A flow is deemed as active, during a certain time

interval, if it always has packets awaiting service. DRR scheduler assigns quantum to each active flow. Suppose the transmission rate of an output link is r . Let n be the total number of flows and let p_i be the reserved rate for flow i . Let $p_{\min}$ be the lowest of these reserved rates. Since all the flows share the same output link, we should ensure that the sum of the reserved rates should be equal or less than the transmission rate of the output link. To make sure that each flow receives service proportional to its guaranteed service rate, the DRR scheduler assigns a weight to each flow. The weight is given by

$$w_i = P_i / P_{\min} \quad (3.2)$$

Note that $w_i \geq 1$. Q_i represent the quantum assigned to flow i and $Q_{\min}$ is the quantum assigned to the flow with the lowest reserved rate. The quantum assigned to flow i , Q_i can be calculated as $w_i * Q_{\min}$. In this way, the quantum assigned to each flow is in proportion of its reserved rate. Each flow has a Deficit Counter (DC), and the counter is incremented in each round as the amount of data the scheduler should try to schedule in the next round, considering the pre-defined service rate. This counter also records the difference between the amount of data actually sent thus far, and the amount that should have been sent. When a packet in the flow is served in the round, the counter is decremented by the amount corresponding to the packet size. If a queue is not able to send a packet in the previous round because its packet size is larger than the value of DC, the counter value is kept and will be used in the next round. Therefore, a queue that received less service than its quantum in a certain round is given an opportunity to receive more service in the next round.

In order that the work complexity of the DRR scheduler is $O(1)$ (that is, the complexity doesn't increase with the number of queues or packets), it is necessary that $Q_{\min}$ should be equal to or greater than the size of the largest packet that may potentially arrive at the scheduler. This can guarantee that the packet size at the head of each queue at the start of its service opportunity will always be less than the sum of the DC value and the quantum value of the flow. Therefore, at least one packet is transmitted from each active flow during each round. Otherwise the per-packet work complexity increase to $O(n)$ since one may encounter a situation in which, even after visiting each of the n flows and examining the respective DC values, no packet is eligible for transmission [25]. DRR requires the knowledge of the size of a packet before making a decision on transmission.

DRR can easily be adapted for guaranteed-rate service scheduling and is the most efficient scheduling algorithm for high speed network for its $O(1)$ computational complexity and low operational cost.

3.3 Discussion

Scheduling algorithms play an essential role in providing per-flow or per-class QoS guarantees in packet data networks. The QoS guarantees include bounded delay, guaranteed bit rate and fair service allocation to contending flows.

In terms of fairness and bounded delay performance, the schedulers based on processor sharing are better than the round-robin schedulers [16]. They are work-conserving in nature and achieve nearly perfect fairness. They also exhibit desirable performance characteristics, such as low delay and bounded short-term unfairness. Unfortunately, they are quite expensive in terms of implementation ($O(\log n)$ complexity, where n is the number of active flows) and incur heavy operational overhead. The

complexity arises from two main sources: updating the virtual clock and sorting the packet tags to schedule the new transmission. These prevent processor sharing scheduling algorithms from being cost-effectively implemented in high speed networks.

The round-robin schedulers divide the time line into allocation cycles and allocate each flow a fraction of the available bandwidth in each cycle. The main attraction of this class of schedulers is their simplicity, timestamp recording and packet sorting are no longer needed. DRR algorithm achieves nearly perfect fairness in terms of throughput, requires only $O(1)$ work to process a packet, and is simple enough to implement in hardware.

In Chapter 5, we will present our scheduling schemes which implement Deficit Round-Robin (DRR) algorithm in Resilient Packet Ring networks.

Chapter 4

4. Modeling and Simulation of MPEG-4 Traffic

With the rapid development of information technology, robust transmission of video over networks is becoming an increasingly important application requirement. In order to reduce the bandwidth requirements, the source video data is compressed before transmission. Between many video compression standards, MPEG-4 is most suitable for Internet and wireless links. It is targeted for low bit rates. Its object-based coding functionalities allow for user interaction with audio-visual objects. The main features of importance are MPEG-4's scalable coding techniques and error-resilient tools. These enhance network utilization and enable MPEG-4 senders to be more responsive to changes in variable network conditions. To enable the study of MPEG4 transmission performance over networks, we develop models and tools for MPEG4 traffic generation based on Transform Expand Sample (TES) methodology. After incorporated into network simulator, simulated traffic could be used as source traffic in the study of our proposed scheduling schemes over ring networks.

4.1 Overview of MPEG-4

MPEG-4 is an ISO/IEC standard developed by Moving Picture Experts Group (MPEG). Although MPEG-4 is a video and audio compression standard, in this thesis we will only focus on MPEG-4 video.

Designed as an adaptive representation scheme that also accommodates very low bit rate application, MPEG-4 is very appropriate for Internet and wireless multimedia applications due to the following features[28][29]:

- 1) It is targeted to provide high video quality at relatively low bit rates. The lowest bit rate can be 5k bps.
- 2) Scalable video coding techniques are introduced into MPEG-4 to provide the varying coding bit rate for the varying channel capacity.
- 3) Many error-resilient tools are incorporated into MPEG-4, which guarantee the correctness of the data in the error-prone channels.
- 4) Object-based coding functionalities allow for interaction with audio-visual objects and enable new interactive applications.
- 5) Face animation parameters can be used to reduce bandwidth consumption for real-time communication applications.

MPEG-4 encoders generate three types of frames: I frames, P frames and B frames. They can be described as follows:

- 1) I frames (intra-coded frames) contain the information that results from encoding a still image. They can be decoded without the need for any other frames.
- 2) P frames (predictively coded frames) require information from previous I frames and/or P frames for encoding and decoding.

3) B frames (bi-directionally predictively coded frames) require information from the previous and following I frames and/or P frames for encoding and decoding.

MPEG-4 scalable video encoder generates multiple sub-streams in contrast to one compressed bit-stream of non-scalable video encoder. One of the compressed sub-streams from the scalable video encoder is the base layer sub-stream, which can be independently decoded and provides coarse visual quality. Other compressed sub-streams, which are called enhancement layer sub-streams, can only be decoded together with the base sub-stream and can provide better visual quality. For example, temporal scalability is a scheme to compress the raw video data into two layers at the same spatial resolution, but different frame rates. The base layer is coded at a lower frame rate. In contrast, the enhancement layer compresses a video with a higher frame rate providing the missing frames. In the base layer only P-type prediction is used, while in the enhancement layer prediction can be either P-type or B-type from the base layer or P-type from the enhancement layer [29].

4.2 Transform Expand Sample (TES) Methodology

TES [10][11] methodology is a versatile methodology for generating data that match closely (in terms of its marginal distribution and auto-correlation function) any set of given observation of time series, such as video traces. We derivate TES models in two phases. First, we generate a correlated sequence with uniform marginal in $[0, 1]$ (also called background TES process) :

$$U_n^+ = \begin{cases} U_0 & n=0 \\ \langle U_{n-1}^+ V_n \rangle & n>0 \end{cases}$$

$$\text{or } U_n^- = \begin{cases} U_n^+ & n \text{ even} \\ \langle 1 - U_n^+ \rangle & n \text{ odd} \end{cases} \quad (4.1)$$

Where U_0 is a random variable uniformly distributed in $[0,1]$. $\{V_n\}$ is a sequence of iid random variables independent of U_0 , called the innovation sequence (with probability density f_v). The operator $\langle \rangle$ denotes the modulo-1 operation which for every real x is defined by

$$\langle x \rangle = x - \max \{ \text{integer}(n) : n \leq x \} \quad (4.2)$$

Both U_n^+ and U_n^- can generate lag-1 autocorrelations in the range $[0, 1]$ and $[-1, 0]$ respectively. It can be shown [10] that the TES background sequence forms a stationary Markov process with a uniform in $[0, 1]$ marginal distribution regardless of the probability law of the innovation density f_v . Since the $\langle \rangle$ operator causes discontinuities, a smoothing operation called the stitching transform is used to remove it. So in the first phase of the TES model development, the objective is to generate a time series of correlated random variables with uniform marginal in $[0,1]$. The amount of correlation depends upon the structure of the density function f_v of the innovation sequence $\{V_n\}$, which in general may be arbitrary. In the second phase, synthetic sample data (the foreground sequence), which resemble the real samples, can be derived from the background sequence using the inversion technique [10][11]. This allows us to transform any uniform random variable to one with an arbitrary distribution. Assumed that a

histogram $H(\cdot)$ of the empirical video sequence has been built consisting of J cells, where the j^{th} cell is positioned on the interval $[l_j, r_j)$ and is characterized by a probability p_j . Such a histogram can be inverted as follows:

$$H^{-1}(x) = \sum_{j=1}^J I_{[C_{j-1}, C_j]}(x) \left[l_j + (x - C_{j-1}) \frac{w_j}{p_j} \right] \quad (4.3)$$

where I_A is the indicator function of the set A , $w_j = r_j - l_j$ is the width of cell j , and

$$C_j = \sum_{i=1}^j p_i \quad 1 \leq j \leq J, (C_0 = 0, C_J = 1) \quad (4.4)$$

The random variable $\{H^{-1}(x)\}_{n=1}^{\infty}$ has a marginal distribution $H(\cdot)$ and autocorrelation that depends upon the innovation density f_v .

The modeling process is carried out in two steps [11]: the first step captures the autocorrelation of the input time series using innovation density function f_v and applying a smoothing technique using the stitching transform [10]. This step generates the background sequence. The second step is to invert the background sequence using the empirical histogram, which always guarantees the matching of the marginal.

The task of finding a suitable f_v that approximates the empirical autocorrelation is carried out via a heuristic search using TES tool [10]. The matching of the empirical

histogram is guaranteed, while the matching of the autocorrelation is carried out in an interactive manner with visual feedback for the best f_v .

4.3 Modeling and Simulation

Traces of MPEG-4 encoded film “STAR WARS IV” [30] are used to present our models and tools for video traffic generation. Due to the periodic nature of the autocorrelation function of the original trace, we first need to separate it into three traces, one for each frame type [11]. This kind of separation also facilitates the development of adaptive rate control function on our traffic generation tools, which we will introduce later. Having done that, we model each of the traces using a separate TES model. Fig. 4. illustrated the models of I frames.

Figure 4. I-frames Model

In Fig. 4, Window 1 (left upper window) shows the sample path of the frame sizes. Window 2 (left bottom window) shows the autocorrelation of the frame sizes. Window 3 (right upper window) shows the histogram of the distribution of the frame sizes. Window 4 (right bottom window) shows the innovation density f_v that was described in Section 3.

To generate the MPEG-4 trace for simulation, we interleave generating samples from the three developed models (I, P and B models) to match the sequence of the original trace (IBBPBBPBBPBBI...)[11]. The final trace generated by interleaving can be compared to the original trace as shown in Fig. 5. Based on the developed models, we also develop adaptive rate control function on our traffic generation tools.

Figure 5. Comparison Between the Original and the Generated Traces

4.3.1 Variable Quantizer Rate Control

The output bit-rate of MPEG-4 encoder can be controlled by applying different quantization table during encoding. Such control, which could be used to respond to congestion conditions in the network, is highly desirable for network performance improvement. In order to study the effects of such a control scheme, a model of the variable quantizer rate controlled MPEG-4 encoder is required.

Fig 6. illustrates the difference when applying different quantization table on I frames of the film “STAR WARS IV”. Through observation, we noticed that the shape of the sample path is shifted, but preserved, while the autocorrelation is remain unchanged. The effect on the sample path of changing the quantization number is equivalent to multiplying the size of each frame by a scalar value. Which also have been shown in H.261 encoded VBR video [31]. This makes it possible to infer frame size for a wide range of quantization numbers from a single table model. The implication is that a model need only be developed for one of the possible quantization tables. Specifying a different quantization table to the encoder is simulated in the source model by applying a scalar multiple to each value generated. Based on this idea, our traffic generation tools simulate the variable quantizer rate controlled MPEG-4 encoder by using basic models and specific numbers of scalars.

Figure 6.Original Traces of I Frames with Different Quantization Table

To demonstrate the usefulness of this approach, we developed a model of same sequence at quantization table 10 by applying a scaling factor to the previously developed I frame model encoded at quantization table 4. Comparison between the developed model and the original trace of I frame at quantization table 10 indicated good agreement, as illustrated in Fig 7.

Figure 7. Comparison between Original Trace of I Frame (Q=10) and Trace Modeled from Q=4.

4.3.2 Variable Layer Rate Control

Multiple layers of streams are generated in MPEG-4 scalable video encoder to accommodate varying channel capacity in wireless applications. Base layer sub-stream can be independently decoded and provides coarse visual quality. Enhancement layer sub-streams can only be decoded together with the base layer sub-stream and provide better visual quality. The scalable video encoder can adjust its output bit rate by sending the base layer sub-stream only or the base layer sub-stream together with enhancement layer sub-streams, according to the feedback of network conditions. To simulate this kind of rate control, we model base layer sub-stream and enhancement layer sub-stream separately according to the TES methodology. When generating the MPEG-4 trace for

simulation, the feedback of the network conditions is used to decide whether interleave or not the trace samples from enhancement layer sub-streams models.

Simulation of a type of temporal scalable encoding is introduced here as an example. In this considered temporal scalable encoding sample, type I and P frames constitute the base layer while the B frames constitute the enhancement layer [32]. We first develop three models each for one frame type. For example, we develop I frame model for all I frames. Then by control of combining B frame model samples, or not, during generation of simulated MPEG-4 traffic, we achieve the variable layer rate control.

Chapter 5

5. New Scheduling Schemes for Resilient Packet Ring Networks

RPR has two kinds of ring architectures: Single Transit Buffer (STB), and Dual Transit Buffer (DTB) [1][2][3]. A high priority and a low priority transmit buffer exist at the client side of RPR. In the STB architecture, a single transit buffer mixes Low Priority (LP) and High Priority (HP) traffic from the transmit buffer into one traffic buffer on the ring. In the DTB architecture, two transit buffers exist for LP and HP traffic. STB has lower implementation complexity.

RPR uses Priority Queue (PQ) [5] as its scheduling scheme. The scheduler handles a specific queue on the basis that all the higher priority queues have been served and emptied. This approach ensures low delay for higher priority classes of traffic. However, it often leads to the starvation of lower priority queues. In the single transit buffer architecture, traffic on the ring, which is a mixture of HP and LP transit traffic, has higher priority over the transmit HP traffic. This could cause the LP traffic on the ring to block the transmit HP traffic from accessing the ring, which is not suitable for the Quality of service (QoS) requirements of HP traffic in terms of delay and delay jitter. For the DTB architecture, overwhelming HP traffic from upstream node, which is not subject to

the fairness algorithm, can block downstream HP traffic from gaining access onto the ring.

To improve the quality of service for high priority traffic transmission, we propose a new RPR scheduling scheme. For the STB architecture, our scheme uses Deficit Round-Robin (DRR) algorithm which alternately selects packets from the single transit buffer and the high priority transmit buffer. For the DTB architecture, DRR algorithm was used to alternately select packets from the high priority transit buffer and the high priority transmit buffer. In both cases, if there is no packet in these two buffers, low priority transmit buffer is then served.

Since DRR only requires $O(1)$ work to process a packet, our scheme does not add extra complexity to the original architecture.

5.1 The proposed scheduling scheme for STB

In the single transit buffer architecture, a transit buffer combines mixed low priority and high priority traffic. The advantage of this configuration is that it simplifies the hardware implementation since it is less complex. However, since the traffic in the transit buffer has the highest priority, even the low priority transit-traffic is served before high priority transmit traffic. During congestion intervals, overwhelming low priority transit traffic can block the transmission of high priority traffic, waiting to gain access onto the ring and degrade the delay and delay jitter performance of packets in the high priority buffer.

5.1.1 STB scheduling scheme

To optimize the video transmission performance of the RPR networks, we proposed a new scheduling scheme for the single transit buffer architecture. In our scheme, we substitute part of the original priority queue scheduling algorithm with deficit round-robin algorithm. We use deficit round-robin scheduling between the transit buffer and the transmit high priority buffer. The transmit low priority buffer is served by the scheduler only when there is no packet in both the transit buffer and transmit high priority buffer,. Fig. 8 shows the MAC architecture of our scheme. Using DRR can guarantee the regular access of the high priority transmit traffic to the ring and only requires $O(1)$ work to process a packet.

Figure 8. The Single Transit Buffer MAC Architecture of the Proposed Scheme.

5.1.2 Simulations of the Proposed Scheduling Scheme for STB

We evaluate the performance of our proposed scheduling schemes in simulation platform OPNET. As a benchmark, we compare the performance of our proposed scheme with the traditional RPR scheduling scheme. We simulate a 10-node ring network in OPNET modeler. The network topology is the same as we demonstrated in Fig.1(Chapter 1). RPR recommended fairness mechanism [1, 2] is used in our simulation. Packets are distributed to emulate real traffic profile, meaning 60% of the packets are 64 bytes, 20% are 512 bytes and 20% are 1518 bytes [33].

For our proposed scheduling scheme, the DRR weight assignment of transmit high priority flow is proportional to its guaranteed service rate. The remaining portion is assigned to the transit buffer. For example, we chose guaranteed high priority transmit traffic rate as 10M bits/s. Since the total link capacity is 100M bits/s, we assign weight 1 to transmit high priority traffic buffer, and weight 9 to its transit buffer.

We designed two scenarios for the simulation study of RPR with single transit buffer. One is Hub scenario, in which traffic from other nodes is forwarded to a common node, the Hub. The other scenario is Mesh scenario, in which all nodes send to all nodes.

5.1.2.1 Hub Scenario

In our simulation, four nodes (6, 7, 8, 9) send packets on the outer ring to the terminal node 0, the Hub. Observations are made in the Hub. Table 1 shows the simulation parameters of the traffic. The generated traffic rate from each node is 33 Mega bits/s, that has 10 Mega bits/s high priority traffic and 23 Mega bits/s low priority traffic. This results in congestion at node 9.

Table 1. Simulation Parameters of Hub Scenario

Parameters	Value
Link Capacity	100 Mbit/s
Fairness Sample Period	100 μ s
Transit Queues Size	1 Mega bits
Packet Sizes	60%: 64 byte 20%: 512 byte 20%: 1518 byte
Traffic Generated in Each Node	HP:10 Mbits/s LP: 23 Mbits/s

If we use the original RPR scheduling scheme, the transit buffer will be emptied first. High priority transmit traffic of node 9 is always queued for serving; even its priority is lower than the low priority traffic from node 6, 7, 8. The delay jitter of node 9 high priority transmit traffic depends on the congestion situation of the single transit buffer.

We compare the proposed scheduling scheme performance in terms of delay and delay jitter with the original RPR scheduling scheme. Fig. 9 and Fig.10 show the end-to-end delay of high priority traffic which demonstrates an average decrease of approximately one hundred times the congested node (Node 9). We also achieve better end-to-end delay for the overall high priority traffic using our proposed scheduling scheme.

Figure 9. The Delay of HP Traffic using the Original RPR Scheduling Scheme (Hub Scenario)

Figure 10. The Delay of HP Traffic Using the Proposed RPR Scheduling Scheme (Hub Scenario)

Figure 11. The Delay Jitter of HP Traffic Using the Original RPR Scheduling (Hub Scenario)

Figure 12. The Delay Jitter of HP Traffic Using the Proposed RPR Scheduling Scheme (Hub Scenario)

The average delay of high priority traffic from the congested node (Node 9) in the proposed scheme always reaches its stable situation (with a maximum variation of 10%) in less than 0.1 second while it is above 0.25 second in the original RPR scheduling scheme. Regarding the high priority traffic from other nodes, the time to reach steady is the same for both schemes.

Fig. 11 and Fig. 12 show the delay jitter of the original system and the proposed one for the high priority traffic. We can see that in all the nodes the delay jitter reach to stable value much faster in our proposed scheme. The improvement is especially more clear for the node 9 (the congested node). Despite the congestion in node 9, the high priority traffic in it still has the chance to pass onto the ring.

By comparing the occupied buffer size of Node 9 using two different schemes, we determine that our proposed scheme needs more transit buffer size. In our simulation, the proposed scheduling scheme needs 25% of the 1 Megabits total transit buffer, while the original scheme occupied 1% of the total transit buffer. The reason is that in the original RPR scheduling, the transit buffer always has the highest priority and empties immediately. However in the proposed scheduling scheme, DRR algorithm serve the transit traffic and high priority transmit traffic alternatively.

We also observed that throughput of high priority traffic in both schemes was the same as the send out traffic.

5.1.2.2 Mesh Scenario

In Mesh (any-to-any) scenario, all 10 nodes in the ring send traffic along the outer ring in a counter clockwise direction. The destination of each packet is uniformly

distributed among all the nodes. Observations are made in one node, node 0. Table 2 shows the simulation parameters of the traffic. The generated traffic rate from each node is 20 Mega bits/s, containing 3 Mega bits/s high priority traffic and 17 Mega bits/s low priority traffic. This results in congestion at each node.

Table 2. Simulation Parameters of Mesh Scenario

Parameters	Value
Link Capacity	100 Mbit/s
Fairness Sample Period	100 μ s
Transit Queues Size	1 Mega bits
Packet Sizes	60%: 64 byte 20%: 512 byte 20%: 1518 byte
Traffic Generated in Each Node	HP: 3 Mbits/s LP: 17 Mbits/s

For Mesh scenario, we also compare the proposed scheduling scheme performance in terms of delay and delay jitter with the original RPR scheduling scheme. Fig. 13 and Fig.14 show the end-to-end delay of high priority traffic. We observed a 10%-20% decrease on end-to-end delay of high priority traffic.

Figure 13 The Delay of HP Traffic Using the Original RPR Scheduling Scheme (Mesh Scenario)

Figure 14 The Delay of HP Traffic Using the Proposed RPR Scheduling Scheme (Mesh Scenario)

Figure 15 The Delay Jitter of HP Traffic Using the Original RPR Scheduling (Mesh Scenario)

Figure 16 The Delay Jitter of HP Traffic Using the Proposed RPR Scheduling Scheme (Mesh Scenario)

Fig. 15 and Fig. 16 show the delay jitter of the original system and the proposed one for the high priority traffic. We recorded delay jitter on average of 0.00025s for simulation using the original scheduling scheme and average of 0.0002s for our proposed scheme. There is approximately 25% decrease on delay jitter of high priority traffic by using our proposed scheduling scheme.

The throughput of high priority traffic in both schemes was the same as the send out high priority traffic.

5.2 The proposed scheduling scheme for DTB

In the dual transit buffer architecture, two transit buffers named low priority transit buffer and high priority transit buffer exist for low priority and high priority traffic. The ring traffic is either dropped if destined for that node, or placed in the high priority or low priority transit buffer, according to the traffic classification. The dual transit buffer allows high priority traffic to have the highest priority and is never blocked by low priority traffic passing through the node.

The scheduling algorithm in DTB scheme is as follows[1][2][4]:

- 1) Packets from the High Priority Transit Buffer.
- 2) Packets from the High Priority Transmit Buffer.
- 3) Packets from the Low Priority Transmit Buffer.
- 4) Packets from the Low Priority Transit Buffer.

In DTB architecture, packets in the high priority transmit buffer have lower priority compared to those in the high priority transit buffer. Thus it needs to wait for the high priority transit buffer to be emptied before it can be served. When there is overwhelming

high priority transit traffic, the high priority transmit buffer will always wait to gain access to the ring. As we know, RPR fairness algorithm only works on the low priority traffic, so we have no effective way to control the passing high priority transit traffic. Under this situation, the high priority transmit traffic will suffer starvation. Since the traffic in high priority transmit buffer such as video and audio is delay sensitive and will be discarded when its delay beyond the time limit, the quality of service will be degraded in heavy high priority transit traffic situation.

5.2.2 DTB scheduling scheme

In order to improve the system performance, we proposed a new scheduling scheme for the dual transit buffer structure. In our scheme, deficit round-robin scheduling algorithm is used to alternately select packets between the high priority transit buffer and high priority transmit buffer. Only when there is no packet in both the high priority transit buffer and the high priority transmit buffer, will the low priority buffer be served by the scheduler according to the following order:

- 1) Packets from the Low Priority Transmit Buffer.
- 2) Packets from the Low Priority Transit Buffer.

Fig. 17 shows the dual transit buffer MAC architecture of our scheme. As the proposed scheme for single transit buffer, using DRR in DTB case also can guarantee the regularly access of the high priority transmit traffic to the ring and it only requires $O(1)$ work to process a packet.

Figure 17 The Dual Transit Buffer MAC Architecture of the Proposed Scheme.

5.2.2 Simulation of the Proposed Scheduling Scheme for DTB

RPR with dual transit buffer gives priority first to the transit high priority buffer, and then the transmit high priority buffer. If there is no packet in both the high priority transit buffer and high priority transmit buffer, low priority traffic will then be served. Thus there is no possibility that low priority transit traffic block high priority transmit traffic accessing onto the ring. However, malicious high priority traffic from the upstream nodes which is not subject to fairness algorithm in RPR, can block the high priority transmit traffic from accessing the ring.

We created a scenario to simulate the overwhelming high priority transit traffic case. The scenario is as follows: All 4 nodes (6,7,8,9) send packets to one terminal node 10, the Hub. Observations are made in the Hub. Table 3 demonstrates the simulation parameters. The generated high priority traffic from node 6,7,8 is 33 Mega bits/s.

Together with the 10 Mega bits/s high priority traffic from node 9, this results in congestion at node 9.

Table 3. Simulation Parameters of DTB

Parameters	Value
Link Capacity	100 Mbit/s
Fairness Sample Period	100 μ s
Transit Queues Size	1 Mega bits
Packet Sizes	60%: 64 byte 20%: 512 byte 20%: 1518 byte
Traffic Generated in Node 6,7,8	HP: 33 Mbits/s LP: 10 Mbits/s
Traffic Generated in Node 9 (congested node)	HP: 10 Mbits/s LP: 10 Mbits/s

If we use the original RPR scheduling scheme, the high priority transit buffer will be emptied first. High priority transmit traffic of node 9 is always queued for serving. In the simulation, total high priority transit traffic from upstream nodes is 99 Mega bits/s. Therefore only 1 Mega bits/s of high priority traffic originated from node 9 can pass to the ring (Fig. 18). Since high priority traffic is not subject to the fairness algorithm, it is not possible to reduce the high priority traffic from the upstream nodes. This will cause the delay sensitive traffic, such as video in high priority transmit buffer, to go over its time limit.

For our proposed scheduling scheme, we assign DRR weight of transmit high priority flow proportional to its guaranteed service rate. In this simulation, we chose guaranteed high priority transmits traffic rate as 10M bits/s. Since the total link capacity is 100M bits/s, we assigned weight 1 to transmit high priority traffic buffer, and weight 9 to its transit buffer. From Fig. 18, we observed that 10 Mega bits/s high priority traffic from node 9 is guaranteed to pass to the ring.

Figure 18 Comparison of Congested Node (node 9) High Priority Transmit Buffer Throughput by Using Different Scheduling Schemes

Chapter 6

6. Evaluation of MPEG-4 Transmission over the new

Scheduling Scheme

Robust transmission of video is a dominant requirement of future application over RPR. In this chapter, we study the performance of delivering MPEG-4 video streams over RPR.

MPEG-4 [34] is the emerging standard for audio and video compression. Being capable of exploiting both spatial and temporal redundancies, it achieves high compression ratio and can encode a video or audio source to almost any level of quality. It distinguishes between three frame types, namely I-, P-, and B-frames. Frames are arranged in Group of Pictures (GOP). A GOP consists of exactly one I-frame and some related P-frames and optionally some B-frames between these I- and P-frames. In MPEG-4, I-frames contain by far the most information. Furthermore, losing an I-frame would cause distortion of all following frames in a GOP. A P-frame loss would only influence the B-frames and the loss of a B-frame would not influence any other frame.

We use the same Hub scenario and parameters as the one we mentioned in section 5.1.2.1. Four nodes (6, 7, 8, 9) send packets on the outer ring to the terminal node 0, the Hub. Observations are made in the Hub. In addition to the traffic mentioned in Table 1 for each node, at node 9 we add an extra MPEG-4 stream traffic source which is also

destined to the Hub. This MPEG-4 stream traffic can be either simulated MPEG-4 stream using TES model we mentioned in Chapter 4 or real MPEG-4 stream. By adding MPEG-4 source at node 9, we can study the transmission performance of MPEG-4 stream at the congested node.

6.1 Delay and Delay Jitter Evaluation

We use two different video sources. “News” with average motion and “Akiyo” with small motion are encoded at 384 Kbps [35]. Both videos are in Common Intermediate Format (CIF), which means they are 352×288 pixel and 30 frames per second. They are encoded according to MPEG-4 with a fixed 12-GOP with two consecutive B-frames (IBBPBBPBBPBB). The UDP protocol is used to segment and packet the MPEG-4 frames. If a frame is larger than 1 k bytes, it is segmented to smaller packets before transmission.

We compared the proposed scheduling scheme performance in terms of delay and delay jitter with the original RPR scheduling scheme. Fig. 19, Fig. 20 and Fig. 21 show the end-to-end delay of I, P and B frames of video sequence “News” for original RPR scheduling scheme and the proposed scheme.

Figure 19. Comparison of End-to End Delay of I frames of Video “News” by Using Different Scheduling Schemes

Figure 20. Comparison of End-to End Delay of P frames of Video “News” by Using Different Scheduling Schemes

Figure21. Comparison of End-to End Delay of B frames of Video “News” by Using Different Scheduling Schemes

We repeated the same simulation on transmitting low motion video “Akiyo”. Fig. 22, Fig. 23 and Fig. 24 show the comparison of end-to-end delay of I, P and B frames of video sequence “Akiyo” respectively by using original RPR scheduling scheme and the proposed scheme.

Figure 22. Comparison of End-to-End Delay of I frames of Video “Akiyo” by Using Different Scheduling Schemes

Figure 23. Comparison of End-to-End Delay of P frames of Video “Akiyo” by Using Different Scheduling Schemes

Figure 24. Comparison of End-to End Delay of B frames of Video “Akiyo” by Using Different Scheduling Schemes

For both videos, we observed better end-to-end delay performance by using our proposed scheme. Table 4. shows the average end-to-end delay comparison of two scheduling schemes. From Table 4, we can see that the average delay of I frames by using original RPR scheduling scheme is almost 212 msec, a large delay in real time applications such as teleconferencing [36]. Using our proposed scheme, the delay for all types of frames is reduced approximately 10 times. The result is more obvious for I frames because I frames are larger in size and are segmented to more UDP packets than other frames.

Table 4. Average End-to-End Delay Comparison

Video Sequence		Average End-to-End Delay (msec)	
		Original Scheme	Proposed Scheme
News	I	212	22
	P	29	3
	B	24	2
Akiyo	I	230	24
	P	30	3
	B	22	2

Fig. 25 shows the delay jitter comparison of I, P, B frames of average motion video “News”. Fig 26 shows the same comparison but based on the video “Akiyo”. Based on our proposed scheme, the delay jitter of video transmission outperforms the original scheme for both videos.

Figure 25 Comparison of Delay Jitter of Video “News” by Using Different Scheduling Schemes

Figure 26. Comparison of Delay Jitter of Video “Akiyo” by Using Different Scheduling Schemes

We observed the occupied high priority transmit buffer size at node 9 using two different schemes. Our proposed scheme requires approximately 2% of the total transmit buffer size, much less than the original RPR scheduling scheme (Fig 27). The original scheme occupied 20 % of the total high priority transmit buffer. This is the reason why our proposed scheme achieves better delay and delay jitter performance for video and other high priority traffic at the congested node. By using the original RPR scheduling scheme, high priority transmit traffic includes video traffic is always blocked by the overwhelming transit traffic at the congested node. However, using our proposed scheduling scheme, high priority transmit traffic can access the ring regularly according to its weight assignment, resulting in only minor traffic waiting in the buffer.

Figure 27 Comparison of Occupied High Priority Transmit Buffer Size of Node 9 (Congested Node) in Two Schemes

We also show the comparison of occupied transit buffer of Node 9 (the congested node) using two different schemes in Fig.28. We observed that our proposed scheme needs more transit buffer size. In our simulation, the proposed scheduling scheme needs 20% of the total transit buffer, while the original scheme occupied 1 % of the total transit buffer. The reason is that in the original RPR scheduling, the transit buffer always has the highest priority and empties immediately. However, in the proposed scheduling scheme, DRR algorithm serve the transit traffic and high priority transmit traffic alternatively. When packets in the high priority transmit are served by scheduler, packets in the transit buffer accumulate. Since there is a fairness control mechanism in RPR, we do not need to worry that traffic in transit buffer will surpass the buffer size limit.

Figure 28 Comparison of Occupied Transit Buffer Size of Node 9 (Congested Node) in Two Schemes

Packets accumulating at the transit buffer are both high priority packets and low priority packets from the upstream nodes. Will this degrade the end to end delay performance of high priority traffic from all nodes? We answer this question by representing, in Fig. 29, the end to end delay of high priority traffic from all nodes. By using our proposed schemes, we observe a decrease of 2 times of end to end delay of total high priority traffic from all the nodes.

Figure 29 Comparison End to End Delay of High Priority Traffic from all Nodes

6.2 Evaluation using PSNR

The subjective quality measures and objective quality measures of real video streams is important because the impression of the user is the ultimate concern [36]. Subjective methods measure the impression of the user watching the video. The human quality impression is given on a scale from 5 (best) to 1 (worst) as in Table 5. This scale is called Mean Opinion Score (MOS). Such subjective methods are described in detail by ITU [37][38], ANSI[39][40] and MPEG[41]. Subjective quality measurement methods are extremely costly: high time consuming, high manpower requirements and special equipments needed.

Table 5. ITU-R Quality and Impairment Scale

Scale	Quality	Impairment
5	Excellent	Imperceptible
4	Good	Perceptible, but not annoying
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very annoying

As subjective methods can often be unaffordable, objective metrics have been developed to emulate the quality impression of the human visual system (HVS) to reduce cost and complexity. In [42] there is a discussion of various objective metrics and their performance compared to subjective tests. The most widespread method is the calculation of peak signal to noise ratio (PSNR) image by image. The PSNR [43] compares the

maximum possible signal energy to the noise energy, which has shown to result in a higher correlation with the subjective quality perception.

To use objective measures in our studies, we adapt EvalVid [36][44] to our simulation platform and calculate PSNR of received video streams at node 0. Evalvid is a toolkit used for the assessment of the video transmission quality. In the evaluation procedure, first the compressed video file is fragmented and transmitted via UDP packets over a real or a simulated network. In order to measure delay, delay jitter, frame loss and frame-by-frame PSNR, the time-stamp and type of each packet transmitted and received are recorded in the sender trace file and receiver trace file. Based on the sender trace file, the receiver trace file and the original encoded video file, EvalVid generates a delay jitter and frame loss report. It also generates a reconstructed erroneous video file, which is subsequently decoded into raw YUV video frames. In the reconstructed erroneous video file, any frame lost during transmission is replaced by the last successfully decoded frame. Finally, the original raw YUV and the reconstructed YUV video frames are used to calculate the frame-by-frame PSNR. The PSNR values obtained from different settings can be used to compare the performance of the different network designs.

Play-out buffer is used at the receiver to reduce frame jitter. The play-out buffer size for video is limited by the perceptual tolerance of the person waiting to view the video. In interactive applications, this limit is about 200ms [45][46]. For non interactive applications, there is no standard approach that has been agreed upon regarding the play-out buffer size [47]. In our simulation, we set the play-out buffer size to 200ms according to interactivity requirement. We consider a frame as lost if its cumulative inter-frame

jitter[48] is beyond the buffer size. This frame will be replaced by the last successfully decoded frame in EvalVid evaluation.

We used two different video conference format sources, “News” and “Akiyo” [49]. Both videos are in Quarter Common Intermediate Format (QCIF), 176×144 pixel and 30 frames per second. The UDP protocol is used to segment and packet the MPEG-4 frames. A frame is segmented to smaller packets before transmission if it is larger than 1 k bytes.

Fig.30 and 31 show the PSNR comparison of the received video streams. From our observation, the quality of video delivered based on our proposed scheme outperforms the original scheme.

Figure 30 Comparison of PSNR of Video “Akiyo” by Using Different Scheduling Schemes

Figure 31 Comparison of PSNR of Video “News” by Using Different Scheduling Schemes

Chapter 7

7. Conclusions and Future Research

7.1 Conclusions

We propose new scheduling schemes for RPR networks for both single transit buffer architecture and dual transit buffer architecture. Using our schemes, traffic in high priority transmit buffers can regularly access the ring according to its round robin weight assignment. This way, the QoS of traffic transmission on the congested node can be significantly improved. Real time video traffic can benefit from such schemes.

Our schemes use the Deficit Round-Robin (DRR) algorithm [7] to substitute part of the recommended priority queue scheduling scheme.

In the single transit buffer architecture, deficit round-robin scheduler alternately serves the transit buffer and the transmit high priority buffer. Only when there is no packet in both the transit buffer and transmit high priority buffer, is the transmit low priority buffer served.

In the dual transit buffer structure, deficit round-robin scheduling algorithm is used to alternately select packets between the high priority transit buffer and high priority transmit buffer. When there is no packet in both the high priority transit buffer and high priority transmit buffer, the low priority buffer is then served by the scheduler.

Since DRR algorithm only requires $O(1)$ complexity, our scheme does not add extra complexity to the original architecture.

We have simulated streaming of MPEG-4 encoded bit-stream from the congested node in an RPR network with single transit buffer in Hub scenario. We investigated the system performance in terms of end-to-end delay and delay jitter of video frames for the original RPR scheduling scheme and that of our proposed scheme. The proposed scheme shows much lower end-to-end delay and lower jitter delay for all types of I, P and B frames of video sequences. The improvement was more significant for I frames, the key frames for video reconstruction at the receiver. In practice, we have limited play-out buffer for video at the receiver. Therefore, we need to discard video frames with delays higher than a fixed threshold. Lower delay and delay jitter of frames will result in lower delays in playing video and smaller play-out buffer requirement. We also use PSNR to objectively measure the video quality after transmission by using different scheduling schemes. By using our proposed scheme, we observe better PSNR average and variance, due to less delay and jitter.

For dual transit buffer architecture, our scheme can ensure the video in high priority transmit buffer transmit at guaranteed bit rates. Thus malicious high priority traffic from the upstream nodes, not subject to a fairness algorithm in RPR, will not block the high priority transmit traffic from accessing the ring.

7.2 Future Research

We can further exploit MPEG-4 scalable video coding techniques in video transmission on RPR. As we know, MPEG-4 has scalable video coding mechanism to provide the varying coding bit rate for the varying channel capacity.

Since RPR use fairness messages to inform the upstream nodes about the network congestion, we can use this message as feedback to the MPEG-4 video source. Therefore, whenever the network experiences sever congestion, MPEG-4 video encoder can use scalable video coding techniques to reduce the output bit rates according to feedback suppress messages. If the MPEG-4 encoder hasn't received any suppress message for a certain time, it can gradually improve its bit rates.

By exploiting the scalable video coding techniques in MPEG-4 encoder, we can reduce the source traffic during network congestion period. This is a better solution than simply discarding packets during transmission.

Bibliography

-
- [1] J. Kao *et al.*, “Darwin Proposal for IEEE Standard 802.17”, Draft 1.0, Jan. 2002.
 - [2] M. Takefman, *et al.*, “IEEE Draft P802.17”, Draft 3.3, Apr. 2004.
 - [3] D. Tsiang and G. Suwala, “CISCO SRP MAC layer Protocol”, Internet Engineering Task Force (IETF) Request for Comments 2892, Aug. 2000.
 - [4] M. J. Francisco, F. Yuan, C. Huang, and H. Peng, “A Comparison of Two Buffer-Insertion Ring Architecture with Fairness Algorithms,” Proceedings of IEEE International Conference on Communications (ICC), May 2003.
 - [5] L. Kleinrock, “Queuing Systems”, First Volume, John Wiley & Sons, New York, 1975.
 - [6] L.Ji, T.N.Arvnitis and S.I.Wooley, “Fair Weighted Round Robin Scheduling Scheme for DiffServ Networks,” IEEE Electronics Letters, Vol.39, No.3, pp.333-335, Feb. 2003 .
 - [7] M. Shreedhar and G. Varghese, “Efficient Fair Queuing using Deficit Round-robin,” IEEE/ACM Trans. Networking, Vol. 4, pp. 375–385, June 1996.
 - [8] J. Zhu ,A.Matrawy, I. Lambadaris and M. Ashourian, “A New Scheduling Scheme for Resilient Packet Ring Networks with Single Transit Buffer” Proceedings of IEEE GLOBECOM workshop, pp.276-280, Nov. 2004.
 - [9] J. Zhu, A. Matrawy, I. Lambadaris and M. Ashourian, “Optimization of Resilient Packet Ring Networks Scheduling for MPEG-4 Video Streaming ”, Accepted in IEEE Ineternational Conference on Communications (ICC) 2005.

-
- [10] B.Melamed, et al., "TES-based Traffic Modeling for Performance Evaluation of Integrated Networks", IEEE INFOCOM, Vol.1, pp.75-84, May 1992.
- [11] A. Matrawy, et al., "MPEG4 Traffic Modeling using the Transform Expand Sample Methodology", Proceedings of IEEE 4th International Workshop on Networked Appliances, pp.249–256, 2002.
- [12] J. Zhu, A. Matrawy and I. Lambadaris, "Models and Tools for Simulation of Video Transmission on Wireless Networks", IEEE CCECE, Vol. 2, pp.781-784, May 2004.
- [13] F. Davik, M. Yilmaz, S. Gjessing and N. Uzun, "IEEE 802.17 Resilient Packet Ring Tutorial", IEEE Communications Magazine, 2004.
- [14] S. Gjessing and F. Davik, "Avoiding Head-of-Line Blocking using an Enhanced Fairness Algorithm in a Resilient Packet Ring", Proceedings of International Conference on Telecommunications(ICT), 2002.
- [15] S. Gjessing and A. Maus, "A Fairness Algorithm for High-speed Networks based on a Resilient Packet Ring Architecture", Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 2002.
- [16] D. Saha, S. Mukherjee and S. K. Tripathi, "Carry-Over Round-Robin: a Simple Cell Scheduling Mechanism for ATM Networks", IEEE/ACM Trans. on Networking, Vol.6, No.6, pp.779-796, Sep.1998.
- [17] A. K. Parekh and R. G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Network: The Single Node Case", IEEE/ACM Trans. Networking, Vol. 1, pp.344–357, June 1993.
- [18] L. Zhang, "Virtual Clock: A New Traffic Control Algorithm for Packet Switching Networks", Proceedings of SIGCOMM, pp.19–29, 1993.

-
- [19] A. Demers, S. Keshav, and S. Shenkar, "Analysis and Simulation of Fair Queuing Algorithm", Proceedings of SIGCOMM, pp.1–12, Sept. 1989.
- [20] S. J. Golestani, "A Self-clocked Fair Queuing Scheme for Broadband Applications", Proceedings of IEEE INFOCOM, pp.636–646, June 1994.
- [21] M.Katevenis, S.Sidiropoulos and C.Courcoubetis, "Weighted Round-robin Cell Multiplexing in a General-purpose AT Switch Chip", IEEE J. Sel. Areas Commun., pp.1265-1279, 1991.
- [22] H. Zhang, "Service Disciplines for Guaranteed Performance Service in Packet Switching Networks.", Proceedings of the IEEE, Vol. 83, No.10, pp.1374-1396, Oct. 1995.
- [23] S.Lam and G.Xie, "Burst Scheduling: Architecture and Algorithm for Switching Packet Video", IEEE INFOCOM, Vol.3, pp.940-950, Apr. 1995.
- [24] L. Kleinrock, "Queueing Systems", Vol.2, Wiley, New York, 1976.
- [25] S. S. Kanhere, H. Sethu and A.B.Parekh, "Fair and Efficient Packet Scheduling Using Elastic Round Robin", IEEE Trans. On Parallel and Distributed Systems, Vol.13, No.3, pp.324-336, Mar. 2002.
- [26] J. Nagel, "On Packet Switches with Infinite Storage," IEEE Trans. on Commun., Vol.35, No.4, pp.435-438, Apr.1987
- [27] S.S.Kanhere and H.Sethu, "On the Latency Bound of Deficit Round Robin" Proceedings of Eleventh International Conference on Computer Communications and Networks, pp.548 – 553, Oct.2002.
- [28] R. Koenen, "Overview of the MPEG-4 Standard", ISO/IEC JTC1/SC29/WG11 M4030, Mar. 2001.

-
- [29] B.Yan and K.W. Ng, "A Survey on the Techniques for the Transport of MPEG-4 Video over Wireless Networks" IEEE Trans. on Consumer Electronics, Vol.48, Issue 4, pp.863 – 873, Nov. 2002.
- [30] Video Traces for Network Performance Evaluation, URL: <http://trace.eas.asu.edu/>, Retrieved Nov. 2004.
- [31] C.M. Sharon et al., "Accurate Modeling of H.261 VBR Video Sources for Packet Transmission Studies", Proceedings of IEEE GLOBECOM, Nov.1995.
- [32] M. Reisslein et al., "Traffic and Quality Characterization of Scalable Encoded Video: a Large-scale Trace-based Study" technical report, Dec.2002.
- [33] IEEE 802.17 Working Group, "Guidance for IEEE 802.17 RPR Performance Simulations," http://www.ieee802.org/17/performance_committee.htm , Nov.2001.
- [34] ISO-IEC/JTC1/SC29/WG11. ISO/IEC 14496: Information technology - Coding of audio-visual objects, 2001.
- [35] URL: <http://www.tkn.tu-berlin.de/~jklaue/cif.html>, Retrieved Nov. 2004.
- [36] J. Klaue, B. Rathke and A. Wolisz, "Supporting Model-Based Design of Systems, EvalVid -- A Framework for Video Transmission and Quality Evaluation" Lecture Notes in Computer Science, Vol. 2794 , pp.255 – 272, 2003.
- [37] ITU-R Recommendation BT.500-10. Methodology for the subjective assessment of the quality of television pictures, March 2000.
- [38] ITU-T Recommendations P.910 P.920 P.930. Subjective video quality assessment methods for multimedia applications, interactive test methods for audiovisual communications, principles of a reference impairment system for video, 1996.

-
- [39] ANSI T1.801.01/02-1996. Digital transport of video teleconferencing / video telephony signals. ANSI, 1996.
- [40] ANSI T1.801.03-1996. Digital transport of one-way video signals - parameters for objective performance assessment. ANSI, 1996.
- [41] ISO-IEC/JTC1/SC29/WG11. Evaluation methods and procedures for mpeg-4 tests, July, 1996.
- [42] S.Wolf and M. Pinson. "Video Quality Measurement Techniques." Technical Report 02-392, U.S. Department of Commerce, NTIA, June 2002.
- [43] M. J. Riley and I.E.Richardson, "Digital Video Communications", Artech House, Norwood, 1997.
- [44] H.C. Ke, "How to evaluate MPEG video transmission using the NS2 simulator?", URL: http://hpds.ee.ncku.edu.tw/~smallko/ns2/Evalvid_in_NS2.htm, Retrieved Nov. 2004.
- [45] H.I.Lee, " Internet Transmission of Real-time MPEG Video", <http://www.mtholyoke.edu/acad/compsc/Honors/Hu-Imm-Lee/>
- [46] P. Brady: "Effects of Transmission Delay on Conversational Behavior on Echo-free Telephone Circuits", Bell System Technical Journal, Vol. 50, No.1, pp.115-134, Jan. 1971.
- [47] A.Aguiar, C. Hoene, J.Klaue, H.Karl, H. Miesmer and A. Wolisz, "Channel-aware Schedulers for VoIP and MPEG4 based on Channel Prediction", In Proceedings of 8th International Workshop on Mobile Multimedia Communications (MoMuC'03), Oct. 2003.

[48] L. Tionardi and F. Hartanto, "The use of Cumulative Inter-Frame Jitter for Adapting Video Transmission Rate", IEEE Tencon, pp.364-368, 2003.

[49] URL: <http://www.tkn.tu-berlin.de/~jklaue/qcif.html>, Retrieved Nov. 2004.