Fall 1-31-2013

Novel algorithms for fair bandwidth sharing on counter rotating rings

Mete Yilmaz
New Jersey Institute of Technology

Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

NOVEL ALGORITHMS FOR FAIR BANDWIDTH SHARING ON COUNTER ROTATING RINGS

by
Mete Yilmaz

Rings are often preferred technology for networks as ring networks can virtually create fully connected mesh networks efficiently and they are also easy to manage. However, providing fair service to all the stations on the ring is not always easy to achieve.

In order to capitalize on the advantages of ring networks, new buffer insertion techniques, such as Spatial Reuse Protocol (SRP), were introduced in early 2000s. As a result, a new standard known as IEEE 802.17 Resilient Packet Ring was defined in 2004 by the IEEE Resilient Packet Ring (RPR) Working Group. Since then two addenda have been introduced; namely, IEEE 802.17a and IEEE 802.17b in 2006 and 2010, respectively. During this standardization process, weighted fairness and queue management schemes were proposed to be used in the standard. As shown in this dissertation, these schemes can be applied to solve the fairness issues noted widely in the research community as radical changes are not practical to introduce within the context of a standard.

In this dissertation, the weighted fairness aspects of IEEE 802.17 RPR (in the aggressive mode of operation) are studied; various properties are demonstrated and observed via network simulations, and additional improvements are suggested. These aspects have not been well studied until now, and can be used to alleviate some of the issues observed in the fairness algorithm under some scenarios. Also, this dissertation focuses on the RPR Medium Access Control (MAC) Client implementation of the IEEE

802.17 RPR MAC in the aggressive mode of operation and introduces a new active queue management scheme for ring networks that achieves higher overall utilization of the ring bandwidth with simpler and less expensive implementation than the generic implementation provided in the standard. The two schemes introduced in this dissertation provide performance comparable to the per destination queuing implementation, which yields the best achievable performance at the expense of the cost of implementation. In addition, till now the requirements for sizing secondary transit queue of IEEE 802.17 RPR stations (in the aggressive mode of operation) have not been properly investigated. The analysis and suggested improvements presented in this dissertation are then supported by performance evaluation results and theoretical calculations. Last, but not least, the impact of using different capacity links on the same ring has not been investigated before from the ring utilization and fairness points of view. This dissertation also investigates utilizing different capacity links in RPR and proposes a mechanism to support the same.

NOVEL ALGORITHMS FOR FAIR BANDWIDTH SHARING ON COUNTER ROTATING RINGS

by
Mete Yilmaz

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Engineering

Department of Electrical and Computer Engineering

January 2013

APPROVAL PAGE

NOVEL ALGORITHMS FOR FAIR BANDWIDTH SHARING ON COUNTER ROTATING RINGS

Mete Yilmaz

Dr. Nirwan Ansari, Dissertation Advisor — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Edwin Hou, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Roberto Rojas-Cessa, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Osvaldo Simeone, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Guiling Wang, Committee Member — Date
Associate Professor of Computer Science, NJIT

BIOGRAPHICAL SKETCH

Author: Mete Yilmaz

Degree: Doctor of Philosophy

Date: January 2013

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2013

- Master of Science in Computer Engineering,
  Bogazici University, Istanbul, Turkey, 1999

- Bachelor of Science in Electrical Engineering,
  Bogazici University, Istanbul, Turkey, 1995

Major: Computer Engineering

Publications:

M. Yilmaz and N. Ansari, "Achieving Destination Differentiation in Ingress Aggregated Fairness for Resilient Packet Rings by use of Weighted Destination Based Fair Dropping," submitted to *Computer Networks*.

M. Yilmaz and N. Ansari, "Resilient Packet Rings with Heteregenous Links," *Proc. The Seventeenth IEEE Symposium on Computers and Communication (ISCC'12)*, Cappadocia, Turkey, July 1 - 4, 2012, pp. 708-712.

M. Yilmaz and N. Ansari, "Weighted Fairness and Correct Sizing of Secondary Transit Queue in Resilient Packet Rings," *IEEE/OSA Journal of Optical Communications and Networks*, Vol. 2, No. 11, pp. 944-951, Nov. 2010.

M. Yilmaz, N. Ansari, J-H. Kao, and P. Yilmaz, "Active Queue Management for MAC Client Implementation of Resilient Packet Rings," *Proc. IEEE International Conference on Communications (ICC 2009)*, Dresden, Germany, Jun. 14-18, 2009, 5 pages.

M. Yilmaz and N. Ansari, "Weighted Fairness in Resilient Packet Rings," *Proc. IEEE International Conference on Communications (ICC 2007)*, Glasgow, Scotland, UK, Jun. 24-28, 2007, pp. 2192 - 2197.

F. Davik, M. Yilmaz, S. Gjessing, and N. Uzun, “IEEE 802.17 Resilient Packet Ring Tutorial,” *IEEE Communications Magazine*, Vol. 42, No. 3, pp. 112–118, March 2004.

To my beloved family

ACKNOWLEDGMENT

I am deeply grateful to Dr. Nirwan Ansari for not only being a great research supervisor by providing insight and intuition but also guiding me and encouraging me patiently for many years. I cannot thank him enough for all the support that he provided during my research. I would also like to thank Dr. Edwin Hou, Dr. Roberto Rojas-Cessa, Dr. Osvaldo Simeone, and Dr. Guiling Wang for serving in my dissertation committee. Special thanks are given to Dr. Necdet Uzun for initiating this long journey and Dr. Attila Fatih Unal for his constant encouragement during my work. I would also like to thank Cisco for financially supporting my PhD. In addition, I would like to thank my family for their continuous support and inspiration. Specifically, I owe many thanks to my loving parents, Kadriye and Erdogan for teaching me to never give up, and to my sister, Elif for encouragement and feedback. My special appreciation is due to my wife, Pinar for a peaceful loving home and reviewing my work countless times. Finally, I would like to thank my daughter, Deniz, who not only brought happiness and joy to my life but also helped me to recognize the urgency of time.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

In today's networks, the transfer rates on a single fiber can reach hundreds of gigabytes per second. In these high-speed networks, simple techniques are desired to control and route the traffic since the processing power does not increase at the same rate as the network capacity.

In ring topologies, the stations benefit from the uniform structure of the ring since each station only needs to decide if a packet is destined to itself or not. These rings are built using several point-to-point connections. When the connections between the stations are bidirectional, rings also allow for resilience (a frame can reach its destination even in the presence of a link failure). A ring is also simpler to operate and administrate than a complex mesh or an irregular network. However, in the traditional optical TDM (time division multiplexing) networks, two rings are deployed where one of the rings is kept as a backup ring. This increases the total cost of the network.

In order to support data traffic on top of TDM based networks, different protocol hierarchies have been developed. Clearly, the additional bytes added by each protocol layer decrease the effective bandwidth of a link. In addition, these extra bytes cause processing overhead at the two ends of the connections. The conflicting and overlapping layers may be present in the protocol stack and result in similar functionalities such as error control being carried out more than once.

As demand to carry more data traffic increases as compared to the voice based traffic, operators have started to utilize TDM based ring networks to transport data traffic.

In order to increase the efficiency of carrying data traffic over TDM networks and to address the other concerns noted above, the IEEE 802.17 Resilient Packet Ring group was formed in 2000 under the umbrella of the IEEE 802 LAN/MAN Standards committee. The initial 802.17 standard [1] was released in 2004, followed by an update in 2006 to support wider spatial awareness and another update to support protected inter-ring connection in 2010 to facilitate resilient connectivity between multiple Resilient Packet Rings. The latest update to the standard was made in 2011 to include additional maintenance requests in the standard.

The standard also incorporates a fairness algorithm to provide fair sharing of ring bandwidth among stations on the ring. Unfortunately, the standard algorithm suffers from decreased utilization under some traffic scenarios which have been discussed widely in the academia. Since the fairness algorithm is already an integral part of the standard, the fixes proposed in the academia are not easily applicable within the context of the standard. Therefore, one aspect investigated in this dissertation is to utilize the mechanisms in different ways to avoid such low network utilization for these traffic scenarios.

Specifically, in this dissertation, the weighted fairness aspects of IEEE 802.17 RPR (in the aggressive mode of operation) are shown through network simulations, and additional improvements are suggested. These aspects have not been well studied until now, and can be used to alleviate some of the issues observed in the fairness algorithm under some scenarios. Also, this dissertation focuses on the RPR MAC Client implementation of the IEEE 802.17 RPR MAC in the aggressive mode of operation, and introduces new active queue management schemes for ring networks to achieve higher

overall utilization of the ring bandwidth with simpler and less expensive implementation than the generic implementation provided in the standard. The two schemes introduced in this dissertation provide performance comparable to the per destination queuing implementation, which is supposed to yield the best achievable performance. Furthermore, the requirements for sizing secondary transit queue of IEEE 802.17 RPR stations (in the aggressive mode of operation) have not been properly investigated. The analysis and suggested improvements presented in this dissertation are then validated by performance evaluation results and theoretical calculations. Finally, the impact of using different capacity links on the same ring has not been investigated before from the viewpoint of ring utilization and fairness. This dissertation also investigates non-uniform links in RPR and proposes a mechanism to support the same.

With these points in mind, an overview of SONET, Spatial Reuse Protocol (SRP), and IEEE 802.17 RPR protocols and discussion of payload efficiency of the transport protocols are covered in Chapter 2. Chapter 3 describes in detail contributions of this dissertation. Specifically, weighted fairness and virtual destination queuing along with active queue management schemes are presented. These schemes can be readily incorporated into the current IEEE 802.17 RPR standard without changes. Specific improvements and exemplar cases such as multi-rate ring are also discussed in Chapter 3. Finally, conclusion is drawn in Chapter 4.

CHAPTER 2

A SURVEY OF PACKET TRANSPORT PROTOCOLS

In the following section an overview of SONET will be provided as a baseline along with the discussion of the protocols that enable transferring of data packets over TDM networks. Next Spatial Reuse Protocol (SRP) and IEEE 802.17 RPR will be introduced. Finally bandwidth efficiency of these protocols will be compared in terms of the additional packet headers and trailers required by the protocols.

2.1 SONET

Synchronous Optical Network (SONET) is a standard [2] for optical transport defined by the Exchange Carriers Standards Association (ECSA) for the American National Standards Institute (ANSI). In short, SONET defines optical carrier (OC) levels and electrically equivalent synchronous transport signals (STSs) for the fiber-optic based transmission hierarchy [3].

As its name reveals, SONET is a synchronous networking technique. Every clock in the system can be traced back to a primary reference clock (PRC). Owing to the synchronous property and its frame structure, SONET can provide a more efficient multiplexing through add/drop multiplexers (ADMs) as compared to the older multiplexing techniques. The multiplexing [4] in SONET is somewhat simpler because of the synchronous network as well as the use of byte interleaving instead of bit interleaving. If some adjustment is needed in the source data, this can be accomplished by the pointers in SONET headers. Low-speed synchronous virtual tributary (VT) signals

are also simple to interleave and transport at higher rates. At low speeds, DS1s are transported by synchronous VT-1.5 signals at a constant rate of 1.728 Mbps. Single-step multiplexing up to STS-1 requires no bit stuffing, and VTs are easily accessed. Another important technique used in SONET is automatic protection switching (APS) [5], which provides fast recovery of the system from failures. However, this method requires the use of spare connections that in effect decreases the utilization of the resources by half. Some of the standard SONET line rates along with their equivalent digital rates are shown in Table 2.1.

Table 2.1 SONET Hierarchy

Optical Carrier (OC)	Electrical Equivalent	Bit Rate(Mbps)	Digital Rate
OC-1	STS-1	51.84	28 DS1s
OC-3	STS-3	155.52	84 DS1s
OC-12	STS-12	622.08	336 DS1s
OC-48	STS-48	2488.32	1344 DS1s
OC-192	STS-192	9953.28	5376 DS1s

SONET uses a basic transmission rate of STS-1 equivalent to 51.84 Mbps. Higher level signals are integer multiples of the basic rate. For example, STS-3 is three times the rate of STS-1 ($3 \times 51.84 = 155.52$ Mbps). An STS-12 rate would be $12 \times 51.84 = 622.08$ Mbps.

2.1.1 Advantages of SONET

Merits of SONET include the following.

- High transmission rates are possible with the standardized SONET systems.
- As compared to pre-SONET systems, it is much easier to drop and insert low-bit rate channels from or into the high-speed bit streams in SONET. It is no longer

necessary to demultiplex and then re-multiplex the entire asynchronous mux structure, which is a complex and costly procedure at best.

- With SONET, network providers can react quickly and easily to the requirements of their customers. The network provider can use standardized network elements that can be controlled and monitored from a central location by means of a telecommunications management network (TMN).

- SONET networks include various automatic back-up and repair mechanisms to cope with system faults. Failure of a link or a network element does not lead to failure of the entire network. These back-up connections are also monitored by a management system.

- SONET is an ideal platform for services ranging from POTS (plain old telephone service), ISDN (integrated services digital network) through data communications (LAN, WAN, etc.), and it is able to handle new, upcoming services such as video on demand and digital video broadcasting via ATM.

- SONET makes it much easier to set up gateways between different network providers, network equipment.

2.1.2 Disadvantages of SONET

While SONET capitalizes on the above advantages, it suffers from the following drawbacks.

- High overhead, due to the frame overhead columns, causes a loss of approximately 6.7 percent of the total bandwidth.

- Half of the bandwidth is also wasted because of APS.

- The system is not self-configuring. Operating costs are high as compared to Ethernet.

- SONET does not have a mechanism to support QoS. Only manual bandwidth adjustments can be carried out. Delay variations (jitter) can occur because of the frame pointer adjustments.

- SONET is based on 8kHz voice synchronized time sample with a frame length of 125μs. This results in fixed frame size which is not very flexible for variable length packet transmission.

2.1.3 SONET Layers

SONET comprises the following five layers. The path, line and section layers are shown in Figure 2.1.

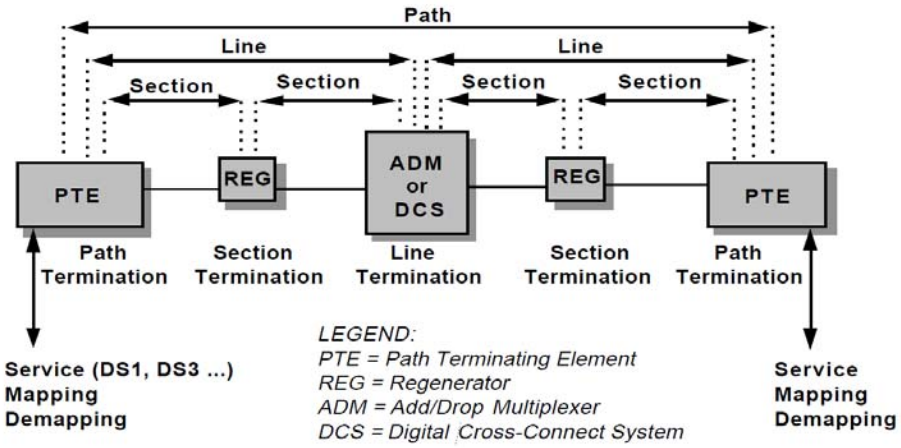

Figure 2.1 Layers of SONET (source: [3]).

- **Photonic Layer**: This layer is the physical layer for SONET, where electrical and optical conversions are carried out.

- **Section Layer:** This layer is responsible for performance monitoring (STS-N signal), local orderwire (channel used by installers to expedite the provisioning of lines), data communication channels to carry information for Operations, Administration, Maintenance, and Provisioning **(**OAM&P) and framing between two ends of a physical connection. The connections are generally regenerator to regenerator or regenerator to multiplexing equipment.

- **Line Layer:** This layer between multiplexing equipment is responsible for performance monitoring of the individual STS-1s, provides express orderwire, data channels for OAM&P, controls the pointer to the start of the synchronous payload envelope (SPE), controls protection switching, failure and alarm signals.

- **Path Layer:** This layer manages the two ends of a connection. It is responsible for performance monitoring of the STS SPE, management of signal label, path status and path trace.

- **VT Layer:** This final layer is used if SPE is used as partitions. Performance monitoring (virtual tributary level) is carried out in this layer. It also provides the signal label, path status, and pointer (depending on the VT type) information.

2.1.4 STS-1 Frame Format

The frame format of the STS-1 signal is shown in Figure 2.2. It is a matrix of nine rows of 90 bytes. The signal is transmitted byte-by-byte beginning with byte one, scanning left to right row by row. The entire frame is transmitted in 125 μs, i.e., 8000 frames are transmitted in every second. The frame transmission time provides compatibility with the voice channels in the telecommunication environment.

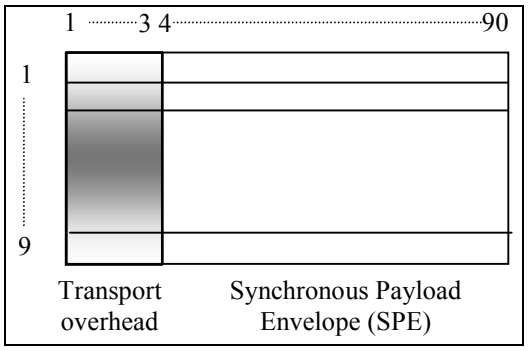

Figure 2.2 STS-1 frame format.

The frame format can be divided into two: the transport overhead and the SPE. The transport overhead is also divided in two, as the section and line overhead. The synchronous payload envelope is also divided into two: the path overhead and payload. Once the payload is multiplexed into the synchronous payload envelope, it can be transported and switched through SONET without having to be examined and possibly demultiplexed at intermediate stations.

The frame is composed of 810 bytes. The transport overhead is 27 bytes and the synchronous payload envelope makes up the rest of 783 bytes. SPE includes not only the payload but also a single column for path overhead and two columns, reserved for some fixed fields. This arrangement leaves a total of 756 bytes for the payload. SPE does not have to start at the first byte of its designated space. Actual starting byte of the SPE is identified by the frame pointer in the transport overhead.

In addition, with respect to the STS-1 base format given in Figure 2.2, it is possible to use sub STS-1 levels. This is accomplished by dividing the SPE into VT (virtual tributary) groups. These VT groups occupy 9 rows and 12 columns of the SPE. This means that there can be seven VT groups at most. VT groups are also subdivided into VTs. Each VT group can accommodate four 1.5Mb channels, or three 2Mb channels, or one 6Mb channel, referred to as VT1.5, VT2, and VT6, respectively. The payload of VTs can be in two modes: static or floating. In the static mode, which is generally used as default, a pointer shows the start of the VT payload inside the VT.

2.1.5 Overheads

2.1.5.1 Section Overhead. The section overhead is the first 9 bytes of the transport overhead, as shown in Figure 2.3.

A1	A2	J0/Z0
B1	E1	F1
D1	D2	D3

Figure 2.3 Section overhead.

It contains A1 and A2 bytes, which uniquely indicate the beginning of the frame. Section trace byte, J0, is used for identifying STM type in the payload (i.e., STM1,

STM3, etc.) or section growth byte Z0 for the other interleaved SONET frames other than the first SONET frame. B1 is an even interleaved parity code for all bits in an STS-N frame. E1 byte is called section orderwire and it is used for voice communication between regenerators, hubs, and remote terminal locations. F1 is a section user channel byte, which terminates at each section terminating equipment. D1, D2, and D3 are used for data communication**s**, which provides a total of 192kbps message channel, used for administration, maintenance, and provisioning.

2.1.5.2 Line Overhead. The line overhead is composed of 18 bytes as shown in Figure 2.4 and occupies the last six rows of the transport overhead of the SONET frame.

H1	H2	H3
B2	K1	K2
D4	D5	D6
D7	D8	D9
D10	D11	D12
S1	M0	E2

Figure 2.4 Line overhead.

The most important bytes that facilitate the seamless multiplexing capabilities of SONET are H1, H2, and H3 pointer bytes. H1 and H2 form a pointer to the beginning of the SPE in the STS frame. However, H3 is used only when a small speed match is necessary between the data source and the SONET clock, i.e., for cases where the data clock is faster than expected and some extra bytes are being accumulated. Then, these bytes can be transferred with the additional data space provided by the H3 byte; this is referred to negative stuffing while the reverse operation is called positive stuffing. D4-D12 are used for line data communication channels, with a total capacity of 576kbps. The

synchronization status byte S1 is used to convey the clock signal quality and clock source. The orderwire byte, E2, is a 64 kbps voice channel used between line entities for an express orderwire.

2.1.5.3 Path Overhead. The format of the path overhead (POH) is shown in Figure 2.5.

J1
B3
C2
H4
G1
F2
Z3
Z4
Z5

Figure 2.5 Path overhead.

It is the first column of SPE. The first byte of the path overhead is J1, an STS path trace byte, to allow the receiving terminal in a path to verify its continued connection to the intended transmitting terminal. B3 is a path bit interleaved parity code byte. C2 is used to indicate the content of the STS SPE. H4 is a **Virtual Tributary (VT)** indicator byte for payload containers. G1 is the path status byte, used to check the performance and status of the path. F2 is a user channel, used for communication between path entities. Z3 and Z4 are used for growth information and Z5 is used for tandem connection monitoring**.**

2.1.5.4 VT Overhead. The VT overhead is part of the VT (Virtual Tributary) and its format is given in Figure 2.6. This overhead enables the communication between the generation point and the destination where the VT is disassembled. The overhead is

distributed over 4 VT frames. The phase of the VT overhead byte is indicated by the H4 byte of the path overhead. V5 is used for performance and error monitoring, signal label and path status. J2 is the signal label. Z6 is tandem connection monitoring. Z7 is the growth byte.

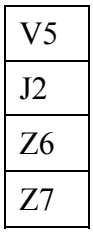

Figure 2.6 VT overhead.

Figure 2.7 shows the construction of an STS-1 frame from different VTs. The interleaving of the bytes can also be seen. The deterministic position of the VT columns enables simpler multiplexing and de-multiplexing capabilities of SONET. A DS1 signal can easily be removed and added from VT1.5. As shown in Figure 2.7, H1 and H2 pointers of the transport overhead point to the payload overhead so that the next point, which will extract the payload information, can locate where the payload is inside the frame. The payload header gives information for the type of the payload and how it is distributed. In this example, the payload is composed of seven VGs, consisting of four VT1.5's (A,B,C and D), four VT2's (X,Y,Z), two VT3's (M,N), and one VT6 (O).

The second row in Figure 2.7 shows the individual VG structure. All bytes from the VTs are interleaved so that only the bytes of a single channel exist in a column of a SONET frame. In the third row of Figure 2.7, the VGs are interleaved to create one column from each one of the four VGs, sequentially, thus forming the payload. This structure helps to remove and add digital signals easily since the location of the channels is fixed in the payload. Obviously, the payload can move back and forth inside every

level of the SONET structure via pointers to compensate for the frequency mismatches between the source and the SONET network.

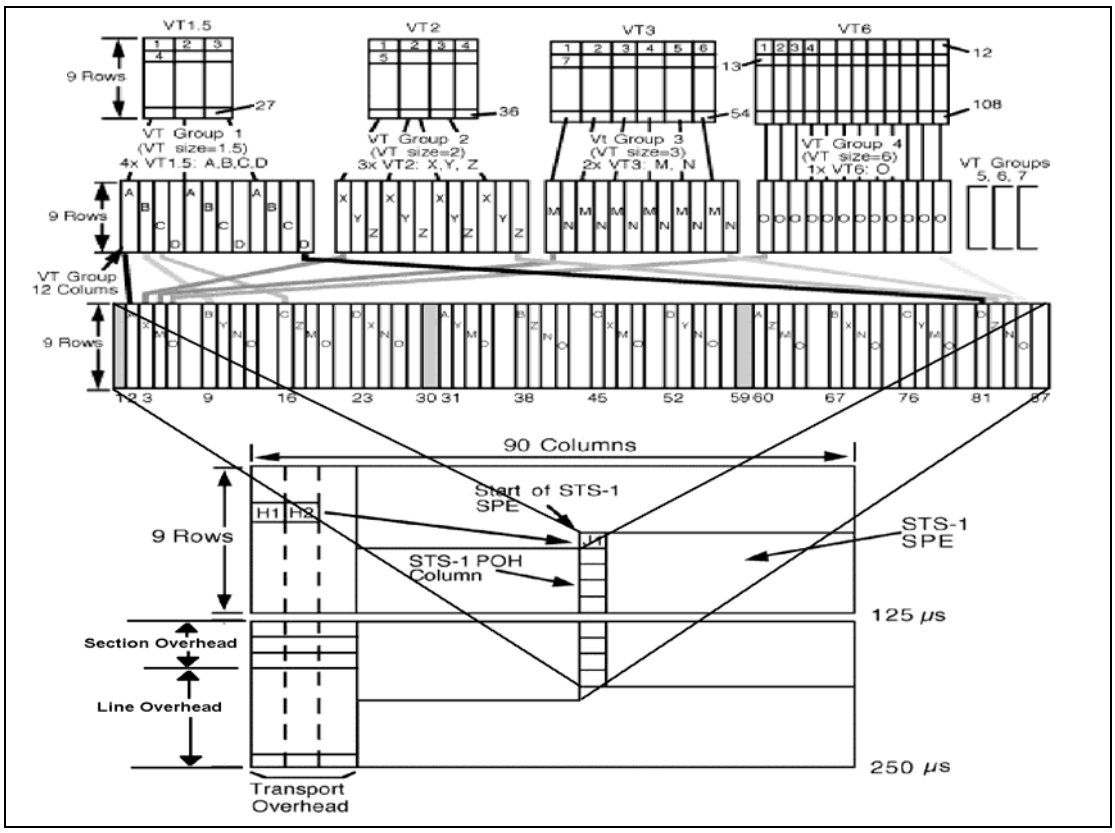

Figure 2.7 SONET STS-1 frame structure (source: [3]).

2.1.6 Automatic Protection Switching

The Automatic Protection Switching (APS) assumes two basic types: linear protection mechanism and ring protection mechanism. The former type controls the point-to-point connections while the latter deals with ring configurations. Both mechanisms use spare connections or components to provide a backup path. The switching between these paths is controlled by the overhead bytes K1 and K2.

Linear Protection: In this configuration, a secondary line is provided. In the simplest configuration, data are transmitted through both lines. If a problem (such as

signal quality degradation) is detected on one line, the communication shifts to the other line. Another choice may be to transmit on a single line. If a problem is encountered, the transmitter and the receiver will switch to the other line. A more economic protection scheme is to reserve a single line for every N active lines for protection. However, this protection scheme cannot overcome two simultaneous signal quality degradations in different connections.

Ring Protection: Ring topologies have greater cost advantages over linear topologies. The protection mechanism for the ring structure can be divided into two. The first protection technique is for unidirectional rings, where the data only flow in one direction on a ring and the protection ring is exactly the same copy of the data ring in the same direction. In a unidirectional ring, if the main connection (fiber ring) has been disconnected, the system will detect the disconnection and will start to use the secondary ring. The second protection technique is for bi-directional rings, where data can flow in any direction on a ring. In each ring/direction, half of the total bandwidth is reserved for data and the other half for protection. If a disconnection occurs in one direction, the stations using the problematic connection will start to retrieve incoming traffic from the other ring. The maximum number of stations in a ring cannot be more than 16 due to the 4-bit node identification field in K1 and K2 bytes of the APS. The protection switching over bi-directional rings can also be provided by pairs of fiber. Each pair of fibers transports working and protection channels, thus resulting in 1:1 protection, i.e. 100% redundancy.

2.1.7 Providing Data Services

When SONET was defined by the communications industry in 1980s, the first consideration was voice communications. In today's networks data communications specifically packet based video traffic is dominating in terms of network utilization. Efficient mechanisms are required to move packets from their sources to their respective destinations. In this section, mechanisms to carry packets over SONET will be investigated.

2.1.7.1 Mapping of ATM Cells. The ATM cells are directly placed into an STS-3c SPE. The cell delineation is carried out with the help of the 5-byte header and the CRC. The standard 155.52Mbps rate is achieved by using a concatenated SONET structure called STS-3c. The start of a cell can also be found from the H4 byte of POH. If there are no packets to transmit, then idle cells will be generated by the adaptation layer. At the receiver side, these idle cells are not transferred to the ATM layer.

2.1.7.2 Point-to-Point Protocol over SONET. The commonly referred term, IP over SONET, is in fact IP over PPP in High Level Data Link Control (HDLC) framing over SONET.

The PPP protocol [6] is a standard for transferring multi-protocol datagrams over point-to-point links. The standard provides a method for encapsulating the multi-protocol datagrams, a link control protocol (LCP) for establishing, configuring and testing the data-link connection, and a family of network control protocols (NCPs) for establishing and configuring different network-layer protocols.

The encapsulation facilitates different protocols to run on the same link simultaneously. The encapsulation format requires additional framing. There are different

methods for framing. The one used by SONET is called HDLC. The encapsulation format is shown in Figure 2.8.

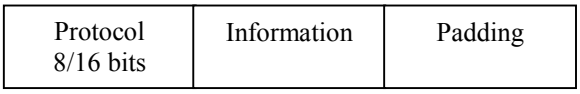

Figure 2.8 PPP Encapsulation.

The fields are transmitted from left to right. The protocol field identifies the datagram type being encapsulated. Although using eight bits is permitted in a compressed format, most of the upper layer specifications prefer the 16-bit protocol header. The protocol field values are standardized in RFC 1340 [7]. The information field can be zero or more octets. The maximum receive unit (MRU) is 1500 octets by default. It is possible to use other values for MRU. The padding field is used to pad the information with octets up to MRU. This is an implementation dependent issue, and it is not necessary to make padding. If it is done, the upper layer protocol should distinguish the information from the padding octets.

LCP is used to automatically agree upon the encapsulation format options, handle varying limits on sizes of packets, detect a loopback link and other common misconfiguration errors, and terminate the link. Peer authentication and link functionality of a link can also be done through LCP.

NCP aims to overcome the problems with the network layer protocols. The two ends of the link and network layer protocols can communicate through this intermediate level and provide end-to-end network layer configuration functionality.

The link operation can be summarized as follows. In order to establish a connection, a channel should be setup first. This is the point where the communication

starts from the “link dead” state. The channel is setup through LCP. The end of this line configuration is signaled with the receipt of Configure_Ack packets on each side. After having agreed upon the parameters (such as MRU), the network layer configuration operations can be carried out. An optional authentication process can be carried out prior to the establishment of the connection. After the previous phases are completed, each network layer protocol is configured through their own NCP. Once all the necessary information is exchanged and the parameters are decided upon, the protocol datagrams can be exchanged. During this connection phase, LCP, NCP and protocol datagrams can be transmitted. The links can be terminated by an LCP terminate request or by a link error. LCP is responsible for informing the upper NCPs of the termination.

2.1.8 PPP Internet Protocol Control Protocol

IP control protocol (IPCP) is an NCP for IP which is defined in RFC 1172 [8]. In order to avoid segmentation and fragmentation, a system should implement the TCP maximum segment size option and MTU discovery. IPCP does not directly provide this information; it allows the IP address information to be exchanged between parties. It also enables or disables the IP compression option.

2.1.9 PPP in HDLC Framing

Over a serial link the PPP encapsulated packets cannot be transmitted directly as it will not be possible to identify the beginning and the end of the packet. The identification of packet start and end can be done in two ways. One way is to use a constant header and trailer for each packet. The other method is to use a mechanism like ATM delineation, which uses CRC to pinpoint the start of a frame. HDLC framing utilizes the former

method. The frame format for PPP in HDLC framing is shown in Figure 2.9 and defined in RFC 1172 [9].

Flag 01111110	Address 11111111	Control 00000011	PPP Encapsulated Datagram	16/32 Bit FCS	Flag 01111110

Figure 2.9 HDLC frame format.

The Flag bytes signal the beginning and ending of the frame. The address field is set to the constant shown value in

Figure 2.9 for PPP framing. In HDLC, the address field is used to address stations; however, only the broadcast address is recognized in PPP. Frames with other than “00000011” in the control field are discarded in PPP. The frame check sequence (FCS) field consists of 16 bits. It is calculated over all bits of the Address, Control, Protocol, Information and Padding fields, not including the Flag and FCS fields.

It is possible to make modifications of the frame structure upon negotiation as well as to remove the control and address information. On reception, the address and control information can be compared with the constant values 0xFF and 0x03. If they do not match, the frame is assumed to be a compressed HDLC frame.

HDLC imposes three specifications according to the line characteristics. The first one is for bit synchronous lines, the second one is for asynchronous lines, and the last one is for octet synchronous transmission lines. PPP over SONET follows the octet synchronous option.

In the PPP encapsulated sequence, octet stuffing is used for HDLC framing. The 0x7d code is the control escape character. Each occurrence of the flag byte or the control escape character in between the start and end flags is replaced with a two octet sequence.

The first byte is the control octet character while the second is the original byte with the 6^{th} bit complemented. This behavior incurs one small disadvantage of HDLC. That is, if a data sequence consists of only 0x7e or 0x7d, the frame to be transmitted will be twice the length of the input sequence.

2.1.10 PPP Over SONET

PPP over SONET (PoS) is a standard method for transporting multi protocol datagrams over SONET point-to-point links. It is defined in RFC 2615 [10]. PoS uses an octet synchronous, full duplex, HDLC like framing. The octet stream is mapped into the higher order VCs. The octet boundaries are aligned with the SONET STS-SPE boundaries. The scrambling of SONET is performed during insertion into SONET STS-SPE to provide adequate transparency and to protect against security threats.

When transmitting an IP datagram, it goes through the sequence of PPP encapsulation first, address and control fields of HDLC are inserted, and then FCS is generated, followed by HDLC framing. The outcome is scrambled with a $1+x^{43}$ scrambler. Finally, the payload is transferred within the SPE of SONET. The path signal label (C2) is set to a hexadecimal value of 0x16 to signal the payload type. If scrambling is left out, the value of hexadecimal 0xCF can be used as the signal label.

None of the compression techniques for protocol, address and control fields are used. One point to note is that the preferred FCS is 32 bit, but 16 bit FCS can be utilized for STS-3c-SPE.

2.1.11 Mapping PPP to STS-1 SPE

Figure 2.10 shows the mapping of PPP packets over SONET STs-1 SPE. Details of this mapping are discussed in [11]. In short, the SPE can start at any point by pointing through the payload pointers. The PPP in the payload is signaled by the C2 byte in the payload overhead. As shown in Figure 2.10, 84 of the 90 columns in a SONET STS-1 frame will be utilized for PPP packets. Alternatively, the VT's can be configured to carry PPP packets with less bandwidth. In addition, concatenated STS-Xc's can be utilized to provide high bandwidth in a similar manner.

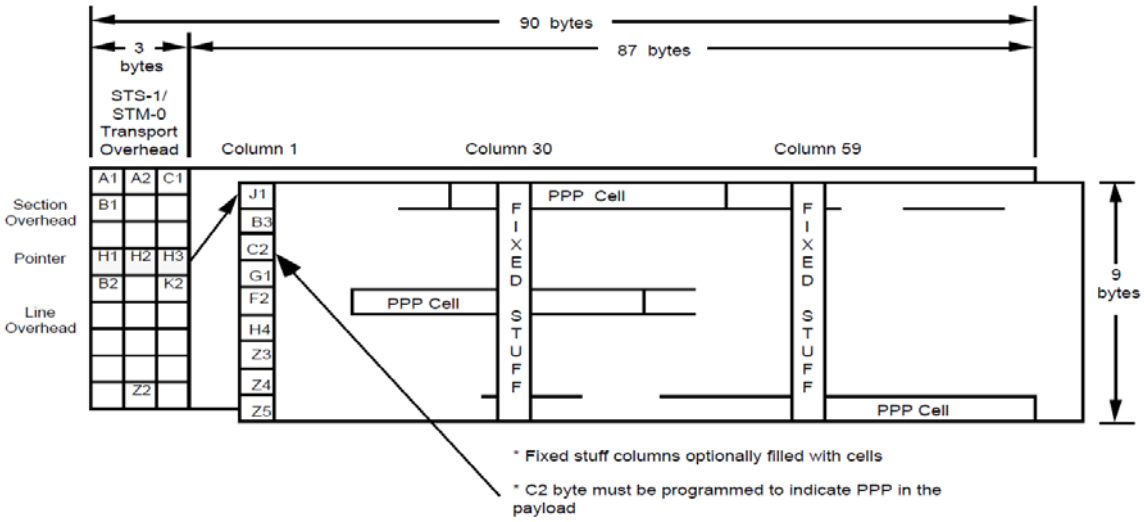

Figure 2.10 A SONET/STS-1 frame with PPP payload (source: [11]).

2.1.12 PPP over Simple Data Link

The overhead in the PoS is not the least that can be achieved. Especially in HDLC framing, certain input will cause an increase in the packet size. In addition, SONET equipment is expensive and over-provisioned for packetized data transmission. Simple data link (SDL), which was introduced in [12], was designed with fewer overheads. RFC

2823 [13] defines the operation over SONET. An Internet draft [14] for operation over raw lightwave channels was also introduced as well, but this draft quickly expired since.

SDL can transmit packets up to 64K bytes in length. Packets can be transmitted without any additional packet length expansion. Link scrambling is possible by using an independent scrambler. The independent scrambler can potentially mitigate malicious user attacks. On the other hand, self-synchronizing scramblers are prone to malicious user attacks. That is, users, who know the standard scrambler characteristics, can feed the system with data which will generate all zero sequences. The SDL implementation also provides a messaging channel for OAM&P. The general frame format for SDL is shown in Figure 2.11.

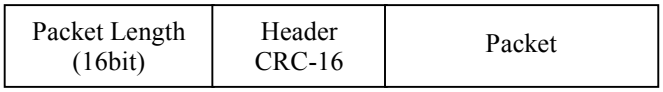

Figure 2.11 SDL framing.

The packet length indicates the length of the packet carried in the Packet field. The full PPP packet size can be used since 16 bits are reserved for the packet size. The header size is fixed to 4 bytes. The start of frame can be detected by an ATM like delineation technique. The CRC is calculated over the header by first setting the header CRC to all zeros. In order to maintain a direct current (DC) voltage level balance, the Packet Length and the Header CRC fields are exclusive OR'ed with the 32 bit value 0xB6AB31E0. The 16-bit CRC provides single error correction and multiple error detection.

The packet portion of the frame is transferred after scrambling, but the header is not scrambled. When the packet length is 0, the distance to the next header will be only 4

bytes and no information will be carried. For packet length field values of one, two and three, the distance to the next header is defined to be 12. These packet length values are used for special SDL messaging. The value 1 is used to transmit the state of the optional set-reset scrambler (with a length of six octets). A 16 bit CRC of the scrambler state is also transferred. The special values 2 and 3 are used for "A" and "B" SDL messages to provide OAM.

The header size in SDL is not limited to 4 and other CRC's can be utilized through messaging. The PPP implementation of SDL does not allow different CRC's in the header as this may incur frame synchronization problems. The header size is fixed to a value of four for PPP over SDL. Furthermore, "A" and "B" messages are not allowed in PPP over SDL implementations. PPP options are set, by default, not to exercise protocol, address and control field compressions. The packet structure for PPP over SDL is defined to be followed by a payload frame check sequence (FCS) with CRC-32. Therefore, the beginning of the next header will be the four offset bytes plus packet length and four bytes of FCS. The overall frame is shown in Figure 2.12.

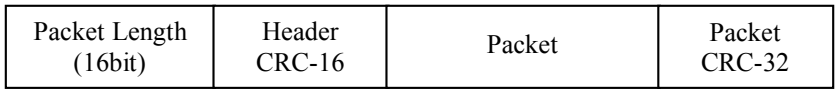

Figure 2.12 PPP over SDL frame structure.

2.1.13 SDL over SONET

The Path Signal Label (C2) as described earlier shows the payload type of SONET. The experimental PSL value of decimal 207 (CF hex) is currently used to indicate that the SPE contains PPP packets using HDLC like framing and transmission without scrambling, and the value 22 (16 hex) is used to indicate PPP with HDLC like framing and transmission with ATM-style X^43+1 scrambling. While HDLC like framing on

SONET/SDH has a fixed seven-octet overhead per frame plus a worst-case overhead of 100% of all data octets transmitted, SDL has a fixed eight octet per frame overhead with zero data overhead. SDL also provides positive indication of link synchronization. The PSL values 0x17 and 0x18 are requested for assignment. The PSL value of 0x17 indicates mapping with a self-synchronous scrambler, and the PSL value of 0x18 indicates mapping with a set-reset scrambler.

Two methods can be used to enable SDL: an LCP-negotiated method and a prior-arrangement method. The former allows easier configuration and compatibility with existing equipment, while the latter allows general use with separate SONET/SDH transmission equipment with PSL limitations.

For the case of LCP negotiation, the LCP Configure-Request messages are transmitted. On reception of LCP Configure-Request with an SDL LCP option or when the peer's transmitted PSL value is received as 23 (or 24), the implementation must shut down LCP, then switch its transmitted PSL value to 23 (or 24), switch encapsulation mode to SDL, wait for SDL synchronization, and then restart LCP by sending an “Up” event into LCP. If the peer does not transmit 23 (or 24), non-SDL O-S PPP encapsulation continues.

When SDL is enabled by prior arrangement, the PSL should be transmitted as either 23 or 24. Any other value may also be used by prior external arrangement with the peer. The values 22 and 207 cannot be used as they are reserved for PPP with HDLC framing. The SDL frames are located within the SPE payload. The frames are allowed to cross SPE boundaries because frames are variable in length.

2.2 Spatial Reuse Protocol

The ring structure, being used in SONET networks, has been used in many other networks such as Token Ring. Spatial Reuse Protocol (SRP) [16] is a MAC layer protocol for ring-based media. SRP alleviates some of the issues in a ring topology. The efficient use of the bandwidth through global and local reuse of the total capacity with a minimal protocol overhead is achieved. The protocol supports two-level priority traffic. The protocol is scalable across a large number of stations with some limitations.

Fairness among the stations is achieved through the SRP Fairness Algorithm (SRP-fa). Protection switching is achieved expeditiously, with speedy switching compatible to SONET. SRP is not a SONET replacement. It aims to provision a cost-effective ring topology as well as performance for packet based networks. It provides two levels of service differentiation by usinga bi-directional ring, which is composed of two symmetric counter-rotating ringlets. This allows wrapping the ring in the event of failure. SRP facilitates a self-configuring network. The stations of the ring structure discover the other stations. For example, after a ring-wrapping, the stations of the network can discover optimized paths, and will then start to use those paths. The SRP packet structure is shown in Figure 2.13.

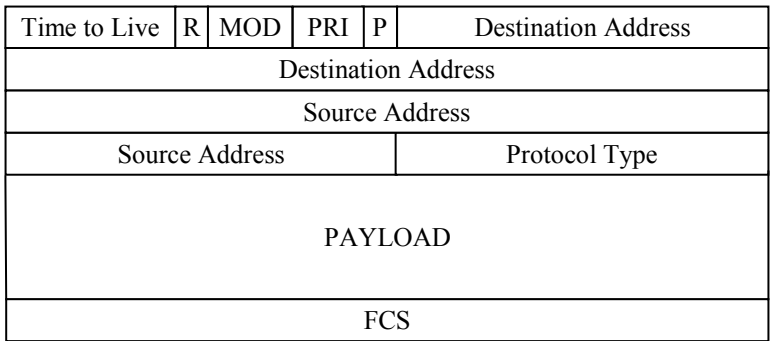

Figure 2.13 SRP packet.

Time-to-live controls the number of hops that a packet can make; this is particularly useful for the case that the destination or source is not able to remove the packet from the ring. The Time-to-live field consists of 11 bits. The R field is a one-bit ring identifier to indicate either the inner or the outer ringer. The priority field is used for providing priority levels among packets. It can support up to eight levels, though two are actually implemented. The mode field (MOD) consists of three bits and is used to identify the type of packet (e.g., data, control or keep alive packet). The single bit parity field is the odd parity value over the last 31 bits of the SRP header. Destination and source addresses reflect 48-bit unique IEEE MAC layer addresses. The Protocol Type field is a two-byte Ethernet type field to reflect the related protocol used to transport the packet.

The efficient use of the ring is accomplished by utilizing a fairness algorithm. Each station has two transit queues. The packet to be transferred is chosen via the fairness algorithm. All high priority traffic is passed to the next station, and if permitted through the information passed in the usage fields of the packets, the low priority traffic will also be forwarded. If congestion occurs in some part of the ring, a feedback mechanism is used to inform the next station, which is transmitting packets to the congested station. This information is then distributed throughout the ring to control the number of packets to be transmitted in every station. This algorithm, therefore, provides a congestion control, and a degree of resource sharing on the ring is accomplished in the steady state. This technology is also referred to as Dynamic Packet Transport (DPT).

2.3 IEEE 802.17 Resilient Packet Ring

In this section, an overview of Resilient Packet Ring based on the published tutorial article [17] is presented. Resilient Packet Ring (RPR, IEEE 802.17) is a ring based network protocol standardized by IEEE [1]. Packet ring based data networks were pioneered by the Cambridge Ring [18], and followed by other important network architectures, notably MetaRing [19], Token Ring [20], FDDI [21], ATMR [22] and CRMA-II [23].

Rings are built by using several point-to-point connections. When the connections between the stations are bidirectional, rings allow for resilience (a frame can reach its destination even in the presence of a link failure). A ring is also simpler to operate and administer than a complex mesh or an irregular network.

Networks deployed by service providers in the Metropolitan Area Networks (MANs) or Wide Area Networks (WANs) are often based on SONET/SDH rings. Many SONET rings consist of a dual-ring configuration in which one of the rings is used as the back-up ring that remains unused during normal operation and utilized only in the case of failure of the primary ring. The static bandwidth allocation and network monitoring requirements increase the total cost of a SONET network. While Gigabit Ethernet does not require static allocation and provides cost advantages; it cannot provide desired features such as fairness and auto-restoration.

Since RPR is standardized in the IEEE 802 LAN/MAN families of network protocols, it can inherently bridge other IEEE 802 networks and mimic a broadcast medium. RPR implements a Medium Access Control (MAC) protocol for access to the

shared ring communication medium, which has a client interface similar to that of Ethernet's.

In the following sections, ring network basics and RPR station design are first discussed, followed by the fairness algorithm, and related issues including topology discovery resilience, bridging, and frame formats.

2.3.1 Ring Network Basics

To transmit unicast packets, frames are added onto the ring by a sender station, which also decides on which of the two counter rotating rings (called ringlet 0 and ringlet 1 in RPR) the frame should take to the receiving station. If a station does not recognize the destination address in the frame header, the frame is forwarded to the next station on the ring. Two transit methods are adopted in RPR: cut-through (the station starts to forward the frame before it is completely received) and store-and-forward.

To prevent frames with a destination address which is not recognized by any station on the ring from circulating forever, the time to live (TTL) field in a frame is decremented as the frame traverses each station on the ring, and the frame is eventually discarded when TTL becomes zero.

When an RPR station is the receiver of a frame, it removes the frame completely from the ring, instead of just copying the contents of the frame and let the frame traverse the ring back to the sender. When the receiving station removes the frame from the ring, the bandwidth, which is otherwise consumed by this frame on the path back to the source, is released for use by other sending stations. This feature is known as spatial reuse.

Figure 2.14 shows an example scenario where spatial reuse is achieved on the outer ring: Station 2 is transmitting to Station 4 at the same time as Station 6 is transmitting to Station 9.

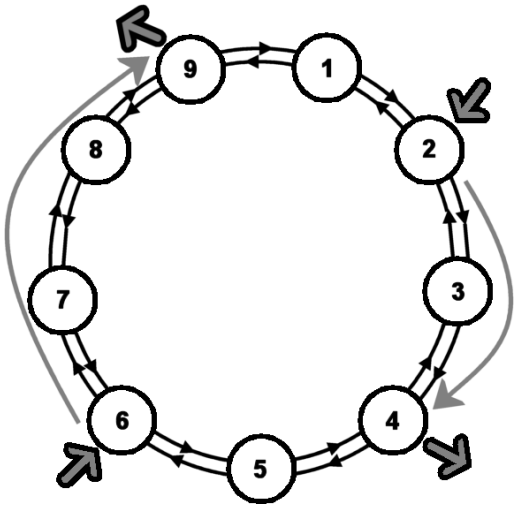

Figure 2.14 Destination stripping and spatial reuse illustrated on the outer ring.

A station's attachment to a ringlet is shown in Figure 2.15. The "insertion buffer" or "transit queue" stores frames in transit while the station itself adds a frame into the ringlet.

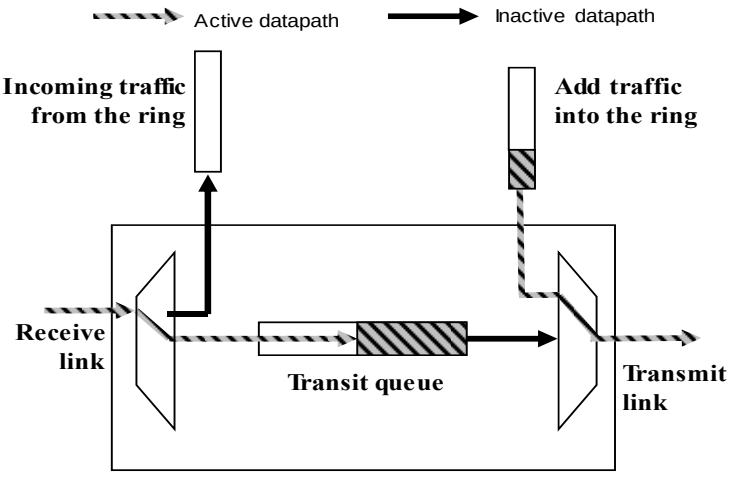

Figure 2.15 Station's attachment to a ringlet.

The ring access method is an important design choice. A token may circulate the stations on the ring so that the station holding the token is the only station allowed to

send packets (like in Token Ring). An alternative access method, called a "buffer insertion" ring, was developed in 1974 [24,25], and utilized later in protocols like MetaRing [19], CRMA-II [23], SCI [26] and SRP [16].

Every station on the ring has a buffer (called a "transit queue", see Figure 2.15) in which frames transiting the station may be temporarily queued. The station must act according to two simple rules. The first rule imposes that a station may only start to add a packet if the transit queue is empty and there are no frames in transit. For the second rule, if a transiting frame arrives after the station has started to add a frame, this transiting frame is temporarily stored (for as long as it takes to send the added frame) in the transit queue.

The above two simple principles obviously need some improvement to make up a complete working protocol that distributes bandwidth fairly. How this is achieved in RPR will be revealed in the next sections.

2.3.2 Station Design and Packet Priority

The stations on the RPR ring implement a medium access control (MAC) protocol that controls the access of the stations to the ring communication medium. Several physical layer interfaces (reconciliation sublayers) for Ethernet (called PacketPHYs) and SONET/SDH are defined. The MAC entity also implements access points that clients can call in order to send and receive frames and status information.

RPR provides a three-level, class based, traffic priority scheme. The objectives of the class based scheme are to provision class A as a low latency, low jitter class, class B as the class with predictable latency and jitter, and finally class C as the best effort transport class. It is worthwhile to note that the RPR ring does not discard frames to

resolve congestion. Hence, when a frame has been added onto the ring, even if it is a class C frame, it will eventually arrive at its destination.

Class A traffic is divided into classes A0 and A1, and class B traffic is divided into classes B-CIR (Committed Information Rate) and B-EIR (Excess Information Rate). The two traffic classes C and B-EIR are called Fairness Eligible (FE) because such traffic is controlled by the “fairness” algorithm, which will be described in the next section.

In order to fulfill the service guarantees for A0, A1 and B-CIR traffic classes, bandwidth needed for these traffic classes is pre-allocated. Bandwidth pre-allocated for class A0 traffic is called "reserved" and can only be utilized by the station holding the reservation. Bandwidth pre-allocated for A1 and B-CIR traffic classes is called reclaimable. Reserved bandwidth not in use is wasted. Bandwidth not pre-allocated and reclaimable bandwidth not in use may be used to send FE traffic.

A station's reservation of class A0 bandwidth is broadcasted on the ring using topology messages (topology discovery is discussed in Section 2.3.4). Having received such topology messages from all other stations on the ring, every station calculates how much bandwidth to reserve for class A0 traffic. The remaining bandwidth, called unreserved rate, can be used for all other traffic classes.

An RPR station implements several traffic shapers for each ringlet that limit and smooth the add and transit traffic. One shaper is tailored for each of the traffic classes A0, A1, B-CIR as well as one for FE traffic. A shaper, referred to as the downstream shaper, is facilitated for all transmit traffic, other than class A0 traffic. The downstream shaper ensures that the total transmit traffic from a station, other than class A0 traffic, does not

exceed the unreserved rate. The other shapers are used to limit the station's add traffic for the respective traffic classes.

The shapers for class A0, A1 and B-CIR are pre-configured, and the downstream shaper is set to the unreserved rate, while the FE shaper is dynamically adjusted by the fairness algorithm.

While a transit queue with the size of one maximum transmission unit (MTU) is enough for buffering frames in transit when the station adds a new frame into the ring; some flexibility for scheduling frames from the add and transit paths can be obtained by increasing the size of the transit queue. For example, a station may add a frame even if the transit queue is not completely empty. Also, a larger queue may store lower priority transit frames while the station is adding high priority frames. The transit queue could have been specified as a priority queue, where frames with the highest priority are dequeued first. A simpler solution, adopted by RPR, is to optionally have two transit queues. Then, high priority transit frames (class A) are queued in the Primary Transit Queue (PTQ) while class B and C frames are queued in the Secondary Transit Queue (STQ). Forwarding from the PTQ has priority over the STQ and most types of add traffic. Hence, a class A frame travelling the ring will usually experience not much more than the propagation delay and some occasional transit delays waiting for outgoing packets to completely leave the station (RPR does not support pre-emption of packets).

Figure 2.16 shows the ring interface with three add queues and two transit queues. The numbers in the circles reflect priorities on the respective transmit links. An RPR station may have one transit queue only (PTQ). In order for class A traffic to move quickly around the ring, the transit queues in all single transit queue stations should then

be almost empty. This is achieved by assigning transit traffic higher priority over all add traffic, and by requiring all class A traffic to be reserved (class A0). Therefore, there will always be room for class A traffic, and class B has priority over class C add traffic, just like in a two transit queue station.

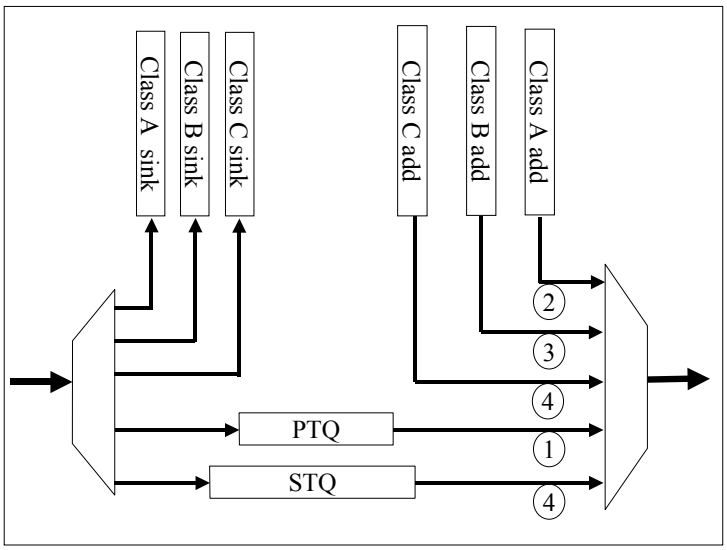

Figure 2.16 The attachment to one ring by a dual transit queue station.

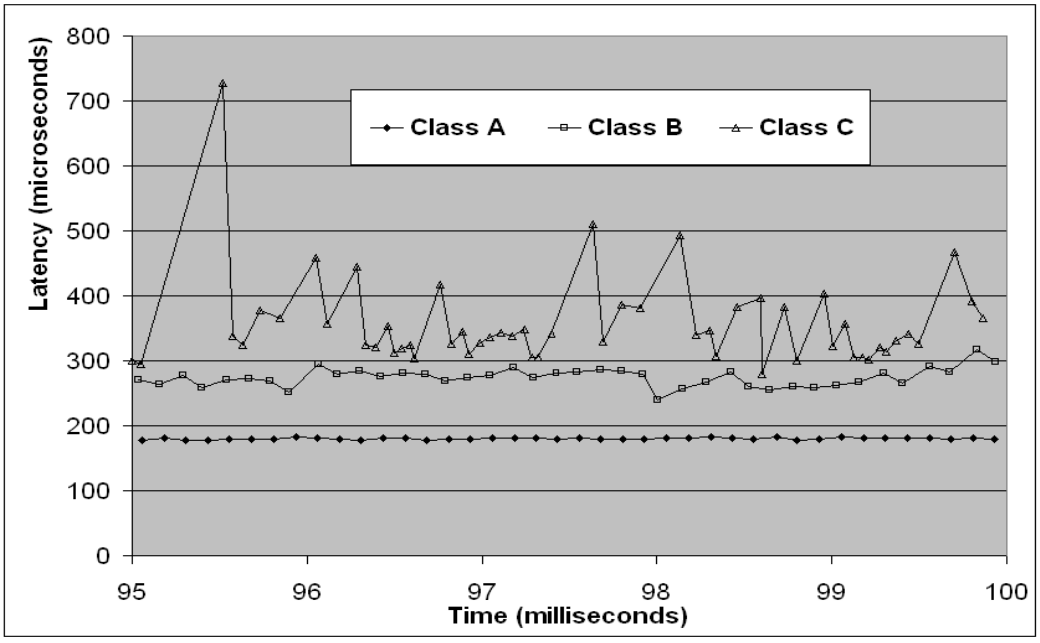

Figure 2.17 Frame latency from station 1 to station 7 on a 16 station overloaded ring.

Figure 2.17 shows an exemplar simulation result where the latency of frames sent between two given stations on an RPR ring is measured. The stations on the ring have two transit queues. The results were obtained by using OPNET [27] based RPR simulation model developed by Cisco during the IEEE 802.17 standardization process. The ring is overloaded with random background, class C traffic. Latency is measured from the time a packet is ready to enter the ring (i.e., first in the add queue) until it arrives at the receiver. Notice how class A traffic keeps its low delay even when the ring is congested. Note that class B traffic still has low jitter under high load while class C traffic experiences rather high delays. Based on the propagation delay, the minimum frame latency is 180 microseconds. An RPR ring may consist of both one transit queue and two transit queue stations. The rules for adding and scheduling traffic are local to the station. Thus, the fairness algorithm works well for both station designs.

2.3.3 RPR Fairness Algorithm

In the basic "buffer insertion" access method, a station may only send a frame if the transit queue is empty. Hence, it is very easy for a downstream station to be starved by upstream ones. In RPR, the solution to the starvation problem is to enforce all stations to behave according to a specified "fairness" algorithm. The objective of the fairness algorithm is to distribute unallocated and unused reclaimable bandwidth fairly among the contending stations, and use this bandwidth to send class B-EIR and class C traffic, i.e. the fairness eligible (FE) traffic.

In defining fair distribution of bandwidth, RPR enforces the principle that when the demand for bandwidth on a link is greater than the supply, the available bandwidth

should be fairly distributed among the contending sender stations. A weight is assigned to each station so that a fair distribution of bandwidth need not be an equal one.

When the bandwidth on the transmit link of a station is exhausted, the link and the station are said to be congested, and the fairness algorithm starts working. The definition of congestion is different for single and dual queue stations, but both types of stations are congested if the total transmit traffic is above certain thresholds. In addition, a single queue station is congested if frames that are to be added have to wait for a long time before they are forwarded, and a dual queue station is congested if the STQ is filling up (and hence transit frames have to wait for a long time before they are forwarded).

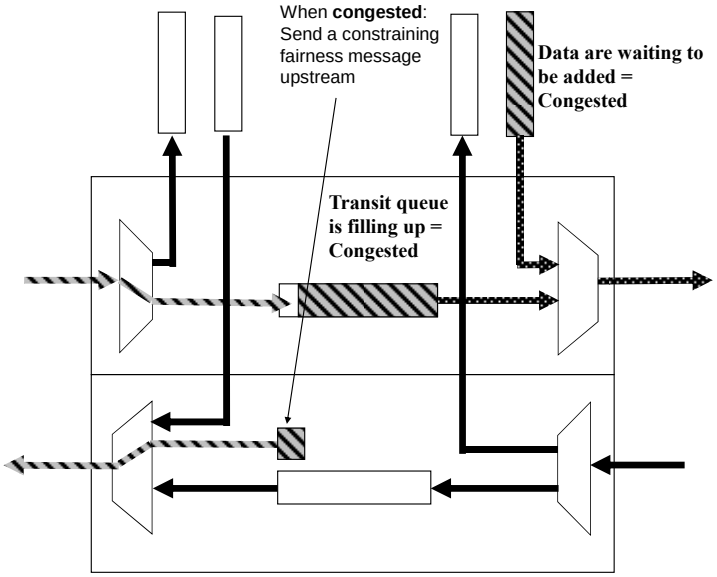

Figure 2.18 Fairness message generation to upstream stations.

The most probable cause of congestion is the station itself and its immediate upstream neighbors. Hence, by sending a so called fairness message upstream (on the opposite ring) the probable cause of the congestion can be reached faster than by sending the fairness message downstream over the congested link. Figure 2.18 shows how the attachment to one ring asks the other attachment to queue and send a fairness message.

The fairness on one ring will be discussed next. The fairness algorithm on the other ring works exactly the same way.

When a station becomes congested, it calculates a first approximation to the fair rate either by dividing the available bandwidth among all upstream stations that are currently sending frames through this station, or by using its own current add rate. This calculated value is sent upstream to all stations that are contributing to the congestion, and these stations have to adjust their sending rates of FE-traffic accordingly. The recipients of this message together with the originating station constitute a congestion domain.

Two options are specified for the fairness algorithm. In the "Conservative" mode, the congested station waits to send a new fair rate value until all stations in the congestion domain have adjusted to the fair rate, and this change is observed by the congested station itself. The estimate of the time to wait (called the Fairness Round Trip Time - FRTT) is calculated by sending special control frames across the congestion domain. The new fair rate may be smaller or larger than the previous one, depending on the observed change.

In the "Aggressive" mode, the congested station continuously (fairness packets are sent with a default interval of 100 microseconds) distributes a new approximation to the fair rate. When the station finally becomes uncongested, it starts sending fairness messages indicating no congestion. A station receiving a fairness message indicating no congestion will gradually increase its add traffic (assuming the station's demand is greater than what it is currently adding). In this way (if the traffic load is stable), the same station will become congested again after a while, but this time the estimated fair rate will

be closer to the real fair rate, and hence the upstream stations in the congestion domain do not have to decrease their traffic rate as much as previously.

2.3.4 Topology Discovery

Topology discovery determines connectivity and the ordering of the stations around the ring. This is accomplished by collecting information about the stations and interconnecting links, via the topology discovery protocol. The collected information is stored in the topology databases of each station.

At system initialization, all stations send control frames, called topology discovery messages, containing their own status, around the ring. Topology messages are always sent all the way around the ring, on both ringlets, with an initial TTL equal to 255 (the maximum number of stations). All other stations on the ring receive these frames, and since the TTL is decremented by one for each station passed, all stations will be able to compute the complete topology image of the network.

When a new station is inserted into a ring, or when a station detects a link failure, it will immediately transmit a topology discovery message. If any station receives a topology message inconsistent with its current topology image, it will also immediately transmit a new topology message (always containing only its own status). Hence, the first station that notices a change starts a ripple effect, resulting in all stations transmitting their updated status information, and all stations rebuilding their topology image.

The topology database includes not only the ordering of the stations around the ring, and the protection status of the stations (describing its connected links, with status signal fail, signal degrade, or idle), but also the attributes of the stations, and the round trip times to all the other stations on the ring.

Once the topology information has become stable, meaning that the topology image does not change during a specified time period, a consistency check will be performed. For example, the station will make sure that the information collected on one ringlet matches the other.

Even under stable and consistent conditions, stations will continue to periodically transmit topology discovery messages in order to provide robustness to the operation of the ring.

When the client submits a frame to the MAC, without specifying which ringlet to use, the MAC uses the topology database to find the shortest path. Information in the topology database is also used in calculating the Fairness Round Trip Time in the conservative mode of the fairness algorithm.

2.3.5 Resilience

As described in the previous section, as soon as a station recognizes that one of its links or a neighbor station has failed, it sends out topology messages. When a station receives such a message indicating that the ring is broken, it starts to send frames in the only viable direction to the receiver. This behavior, which is mandatory in RPR, is called steering.

The IEEE 802 family of networks provision a default packet mode, called “strict” in RPR. This means that packets should arrive in the same order as they are sent. To achieve in-order delivery of frames following a link or station failure, all stations stop adding packets and discard all transit frames until their new topology image is stable and consistent. Only then will stations start to steer packets onto the ring.

The time it takes for this algorithm to converge, that is, from the time the failure is observed by one station until all stations have stable and consistent topology databases and can steer new frames, is the restoration time of the ring. The RPR standard mandates the restoration time to be below 50ms. To accomplish this goal, several design decisions must be considered, including ring circumference, the number of stations, and speed of execution inside each station.

RPR optionally defines a packet mode called “relaxed”, implying that it is tolerant to out-of-order delivery of packets. Such packets may be steered immediately after the failure has been detected and before the database is consistent. Relaxed frames will not be discarded from the transit queues either.

When a station detects that a link or its adjacent neighbor has failed, the station may optionally wrap the ring at the break point (called “wrapping”) and immediately send frames back in the other direction (on the other ringlet) instead of discarding them. Frames not marked as eligible for wrapping are always discarded at a wrap point.

2.3.6 Bridging

RPR supports bridging to other network protocols in the IEEE 802 family and any station on the ring may implement bridge functionality. Transporting Ethernet frames over RPR can provide resilience and class of service support.

RPR uses 48-bit source and destination MAC addresses in the same format as Ethernet (see Section 2.3.4.7). When an Ethernet frame is bridged into an RPR ring, the bridge inserts RPR related fields into the Ethernet frame. Similarly, these fields will be removed if the frame moves from RPR (back) to Ethernet. An extended frame format is

also defined in the standard for transport of Ethernet frames. In this format, an RPR header encapsulates Ethernet frames.

When participating in the spanning tree protocol, RPR is viewed as one broadcast enabled subnet, exactly like any other broadcast LAN. The ring structure is then not visible, and incurs no problem for the spanning tree protocol. The spanning tree protocol may not break the ring, but may disable one or more bridges connected to the ring.

RPR implements broadcast by sending the frame all around the ring, or by sending the frame half way on both ringlets. In the latter case, the TTL field is initially set to a value so that it becomes zero, and the packet is removed when it has travelled half of the ring. Spatial reuse is not achieved by using broadcast.

Since RPR can bridge to any other Ethernet, for example, Ethernet in the First Mile (EFM), Ethernets spanning all the way from the customer into the Metropolitan or even Wide Area Network are envisioned. Whether such large and long ranging Ethernets will be feasible or practical in the future remains to be seen.

Another way to connect RPR to other data networks is to implement IP or layer 3 routers on top of the MAC clients. In this way, RPR behaves exactly like any other Ethernet connected to one or more IP routers. Such IP routers should, in the future, also take advantage of the class based packet priority scheme defined by RPR when they send Quality of Service constrained traffic over RPR.

2.3.7 Frame Formats

Data, fairness, control and idle frames are the four different frame formats defined in the RPR standard. The following subsections introduce the important fields of these frames.

2.3.7.1 Fairness Frames. The 16-byte fairness frame mainly provides the advertised "fairRate" and the source of the fairness frame. The information is used in the RPR fairness algorithm.

2.3.7.2 Control Frames. A control frame is similar to the data frame, but is distinguished by a designated "ft" field value, and its controlType field specifies the type of information carried. There are different types of control frames in RPR, for example, topology and protection information and OAM (Operations Administration and Maintenance).

2.3.7.3 Idle Frames. Idle frames are utilized in order to compensate for rate mismatches among neighboring stations.

2.3.7.4 Data Frames. Data frames have two formats: basic and extended. The basic data frame format is shown in Figure 2.19.

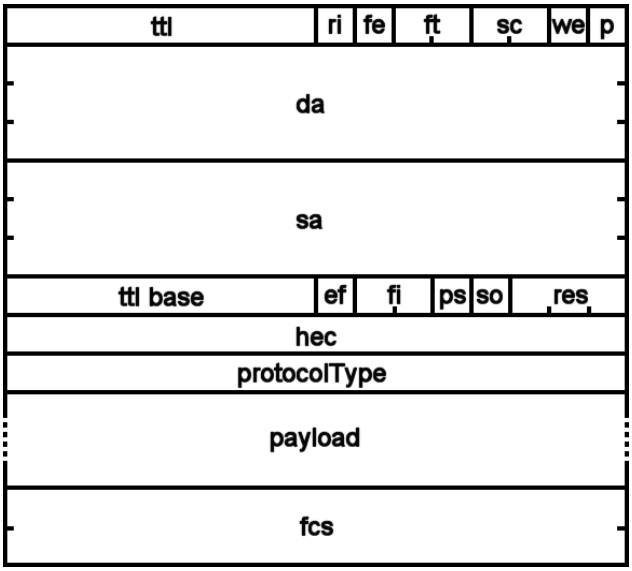

Figure 2.19 RPR basic data frame format.

The Extended frame format is aimed at transparent bridging applications to allow easy egress processing and ingress encapsulation of other medium access control (MAC)

frames. Using the Extended frame format also enables RPR-rings to eliminate out of ordering and duplication of bridged packets. The Extended frame format is not described in this dissertation, and readers are referred to [1] for further details. Table 2.2 provides a short summary of the RPR basic data frame fields.

Table 2.2 RPR Frame Field Definition

Field Name	Description
ttl	The one-byte "time to live" field.
ri	The "ring identifier" bit defines which ringlet the packet was inserted into initially.
fe	The "fairness eligible" bit indicates that the packet has to abide by the rules of the fairness algorithm.
ft	The two bit "frame type": Data, Fairness, Control, Idle.
sc	The two bit "service class": A0, A1, B, C.
we	The "wrap eligible" bit defines if the frame can be wrapped at a wrap station.
p	The "parity" bit is reserved for future use in data frames.
da	The six-byte "destination address".
sa	The six-byte "source address".
ttl base	This field is set to the initial value of the "*ttl*" field when the packet was initially sourced into the ring. It is used for fast calculation of the number of hops that a packet has travelled.
ef	The "extended frame" bit, indicating an extended frame format.
fi	The two bit "flooding indication" is set when a frame is flooded and if so, on one or both ringlets.
ps	The "passed source" bit is set when passing its sender on the opposing ring after a wrap. The bit is used in detecting an error condition where a packet should have been stripped earlier.
so	The "strict order" bit, if set, identifies that the frame should be delivered to its destination in strict order.
res	A three-bit reserved field.
hec	The two byte "header error correction" field protects the initial 16 bytes of the header.

2.4 Bandwidth Efficiency

The protocols utilized to carry packets over SONET require headers and trailers to be added on top of the actual packet. These additional headers and trailers decrease the overall utilization of the link bandwidth. For example, PoS incurs an overhead of two bytes at PPP or six bytes at the HDLC layer. In addition SONET frame also has additional overhead incurred by path and line layers. For example, at OC3/STM-1 rate, there are 90 overhead bytes for each 2430 bytes of SONET frame. Therefore, the efficiency of PoS for a packet which is N bytes long would be :

$$\text{PoS Efficiency (\%)} = \frac{(2430-90)}{2430} \times \frac{N}{(N+2+6)} \times 100 \ ,$$

The above equation does not consider the fact that HDLC is data dependent. Hence, it represents the maximum achievable efficiency.

The SDL incurs a packet framing header of 4 bytes, a PPP header of two bytes, and a trailer of four bytes. It is also possible to use SDL without SONET. Thus, the overhead efficiency of SDL can be expressed as follows:

$$\text{SDL Efficiency (\%)} = \frac{N}{(4+2+N+4)} \times 100 \ .$$

For the SRP, the frame structure incurs 20 bytes of additional overhead, and thus the SRP overhead efficiency is:

$$\text{SRP Efficiency (\%)} = \frac{N}{(N+20)} \times 100 \ .$$

In general, SRP is deployed over SONET with HDLC like framing. Consequently, the overall efficiency will be worse than the POS efficiency.

$$\text{SRP Net Efficiency (\%)} = \frac{(2430-90)}{2430} \times \frac{N}{(N+20+6)} \times 100 \ .$$

The efficiency of the RPR frame is similar to the SRP frame with a little bigger overhead.

$$\text{RRP Efficiency}(\%) = \frac{(2430-90)}{2430} \times \frac{N}{(N+24+6)} \times 100 \ .$$

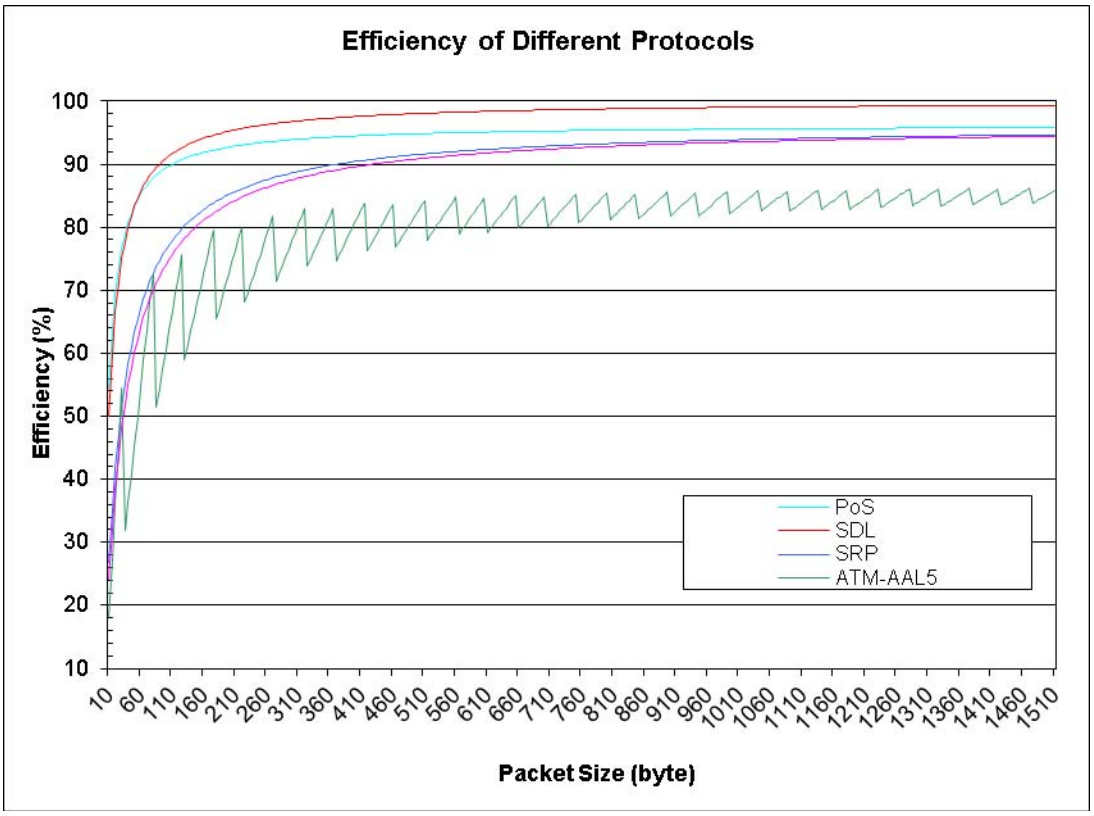

Figure 2.20 Efficiency of PoS, SDL, SRP, and AAL5.

Figure 2.20 displays the efficiency with respect to the packet size. The performances can be compared to ATM Adaptation Layer 5 (AAL5) by using the following formula.

$$\text{AAL5 Efficiency}(\%) = \frac{N}{1.03846 \cdot (\lceil (8+N+8)/48 \rceil \times 53)} \times 100$$

Note that not all packets in the network are small 10-byte packets. Naturally, the packet length distribution plays an important role in overall protocol overhead efficiency. Table 2.3 illustrates the IP packet distribution based on a generic internet packet mix. In

Table 2.3 for each packet size N corresponding efficiency of each protocol is shown. Note that only the first five longest packet lengths were provided for the packet distribution. The first five longest packet lengths contribute to the 77.6 per cent of the total bandwidth. For the packet lengths that contribute to the remaining 22.4 per cent we will assume that the packet size distribution is uniform. For the remaining portion, the efficiencies are combined and provided in the last row of Table 2.3. Finally, Table 2.4 presents the overall expected efficiency for the corresponding protocols based on the traffic distribution provided in Table 2.3.

Table 2.3 IP Packet Distribution and Protocol Efficiencies

IP Packet Size N (byte)	Total Bytes (%)	Total Packets (%)	PoS Eff. (%)	SDL Eff. (%)	SRP Eff. (%)	RPR Eff. (%)	ATM-AAL5 Eff. (%)
1500	48.7	11.5	95.8	99.3	94.7	94.4	85.2
552	15.8	10.1	95.0	98.2	92.0	91.3	83.6
576	7.9	4.9	95.0	98.3	92.1	91.5	80.5
44	4.4	6.1	81.9	81.5	60.5	57.3	40.0
40	0.8	38.9	80.7	80.0	58.4	55.0	36.3
Remaining	22.4	28.5	94.1	97.1	90.1	89.3	79.5

Table 2.4 IP Data Network Efficiency over Different Protocols

PoS Efficiency	SDL Efficiency	SRP Efficiency	RPR Efficiency	ATM-AAL5 Efficiency
94.48	97.62	91.18	90.57	80.87

CHAPTER 3

IMPROVEMENTS OVER SRP AND CONTRIBUTIONS TO THE IEEE 802.17 STANDARD

This chapter presents contributions to the standardization of RPR especially related to fairness in terms of the analysis of the fairness algorithm through simulations [28, 29, 30, 31] that have resulted in three patents [32, 33, 34]. Moreover this chapter provides extensive simulations and substantial improvements that were developed after RPR was standardized. Specifically weighted fairness definition with destination differentiation in Section 3.2, correct sizing of secondary transit queue for underflow case in Section 3.7, MAC client active queue management mechanisms in Sections 3.8 and 3.9 and finally supporting heterogeneous links in Section 3.10 will be introduced in this chapter.

3.1 Implementation of Station Weights in SRP

A station will receive the minimum of equal-share of the link bandwidth or the maximum achievable bandwidth it has over that link. A few different algorithms have been developed to provide such fairness over a ring including SRP, RPR, and distributed virtual-time scheduling in rings (DVSR) [35]. Instead of being limited to an equal share, it is possible to provide specific stations with more bandwidth. This can be accomplished by assigning coefficients to stations and adjusting the usage and allowed usage values with respect to these coefficients. Each station will then get its share of the bandwidth in proportion to the assigned coefficient. The main advantage of the algorithm shown here is that each station does not need to track the coefficient of the other stations. This is

accomplished by normalizing the usage messages before they are transferred on to the ring and readjusting the received usage messages before they are used by the station. Even though the suggested algorithm does not preclude the use of coefficients less than 1, it might be more advantageous not to do so. As long as the coefficients are greater than 1, the usage parameter that will be circulating around the ring will always be limited by an upper limit. This will allow an easier implementation by reserving a fixed number of bits for the usage parameter in fairness messages. This proposed [28] algorithm is now utilized in IEEE 802.17 RPR. When a usage packet from Station *k*-1 is received with values {u, u_{max}}, Station *k* does the following:

```
INIT: a_k = a_max ; u_k = 0
IF (station congested) AND (u = NOT null)
    a_k = (u_max_k/ u_max)u
    u = min {u_k, (u_max_k/ u_max)u}
ELSE IF (station congested) AND (u = null)
    a_k = (u_max_k/ u_max)u_k
    u = u_k
ELSE – station not congested
    a_k = u
    u = null
```

u: usage value received from downstream station

$\mathbf{a_{max}}$: maximum allowed usage

$\mathbf{u_{max\_k}/u_{max}}$: the provisioned weight of station k

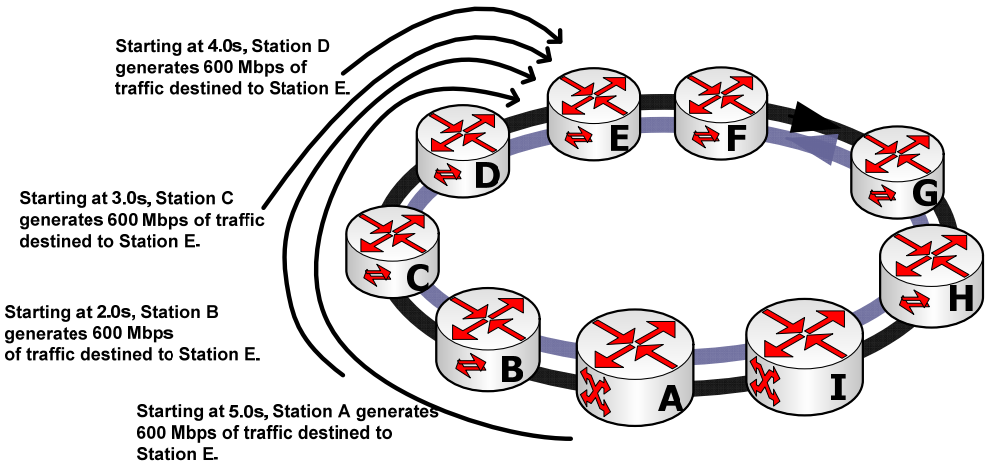

Figure 3.1 Weighted fairness example scenario with equal weights.

Figure 3.1 shows the traffic scenario of a SRP ring with nine stations. Each link has a capacity of 622Mbps. The SRP algorithm is modified to include weighted fairness. Stations A, B, C and D are sending packets to the destination Station E. When all the stations have the same provisioned weights, the behavior is essentially the same as what it will be with the original SRP fairness algorithm. This case is shown in Figure 3.2.

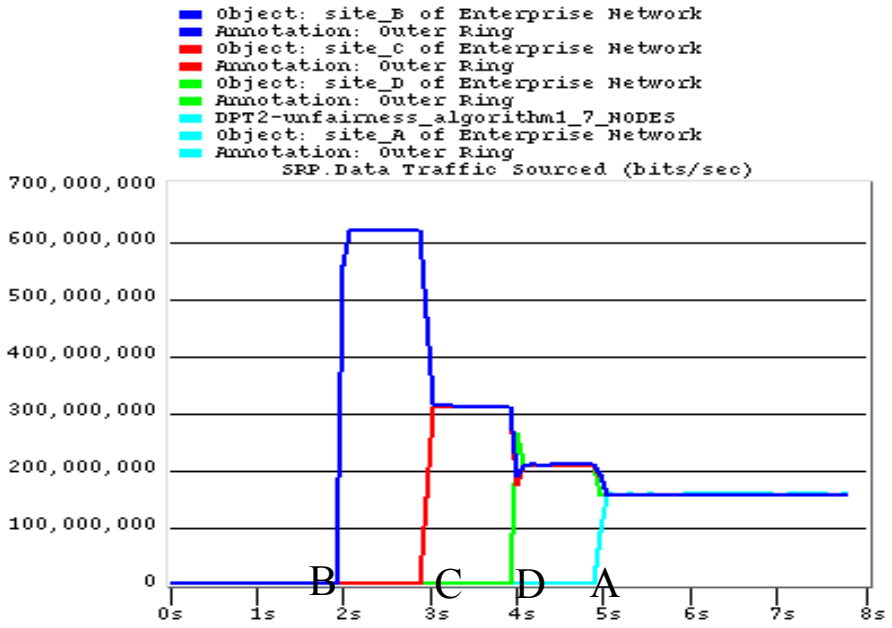

Figure 3.2 Bandwidth distribution of the stations on the ring with equal weights.

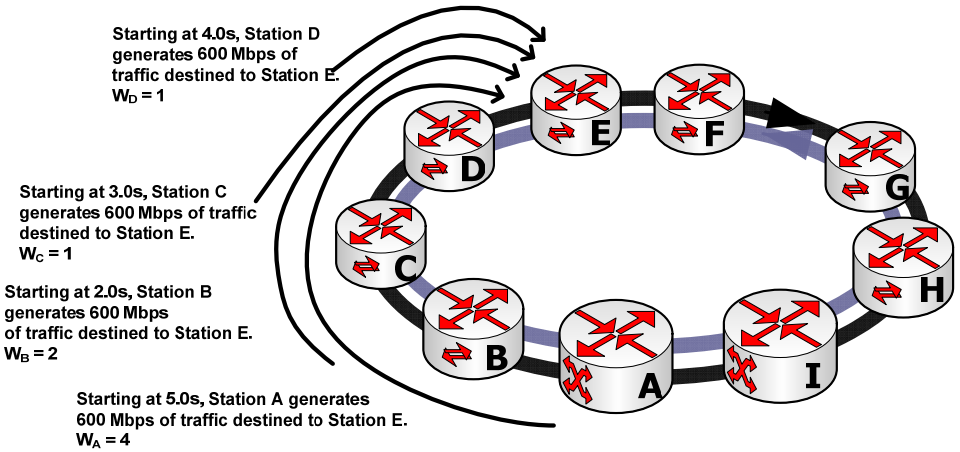

Figure 3.3 Weighted fairness example scenario with different weights.

Another example with different assignment of weights is shown in Figure 3.3. In this scenario, Stations A, B, C, and D are assigned with weights of 4, 2, 1, and 1, respectively. The results of this scenario are shown in Figure 3.4. Stations A, B, C, and D will be allowed to source 311Mbps, 155.5Mbps, 77.75Mbps and 77.5Mbps of traffic, respectively.

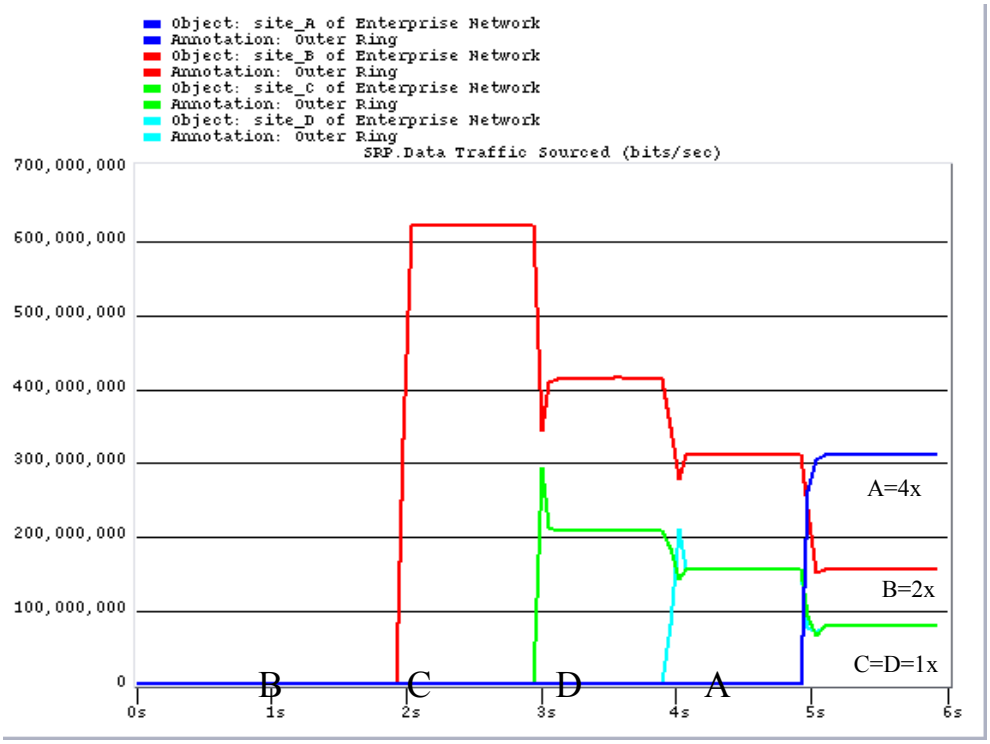

Figure 3.4 Bandwidth distribution of the stations on the ring with different weights.

3.2 Weighted Bandwidth Distribution in RPR

The following section is based an earlier contribution [36] which provided the definition of weighted fairness in RPR and discussed a deficiency of the fairness algorithm as well as provided useful scenarios where weighted fairness can be utilized. However, the definition presented in this section is an enhanced version as compared to the definition presented in [36]. The definition provided in this section allows destination differentiation via weights for each flow. Note that the fairness aspects of RPR have been

investigated in depth in the light of interesting scenarios as those described in [35], [37], and [38]. Improvements for the current fairness algorithm of the IEEE 802.17 have also been proposed as reported in [35], [37], [39], and [40]. However, the weighted aspect of the fairness algorithm has not been thoroughly investigated prior to the publication of the work [36] presented at ICC in 2007.

3.2.1 Weighted Ring Ingress Aggregated Fairness with Destination Differentiation

The objective of the fairness algorithm is to distribute the unallocated bandwidth around the ring among stations in a fair manner. In the case of Figure 3.5 (assuming that all the stations have equal weights), Stations *3* and *4* will get an equal amount of the link bandwidth (the link between Stations *2* and *3*), while Stations *1* and *2* will get an equal amount of the link bandwidth (the link between Stations *1* and *7*).

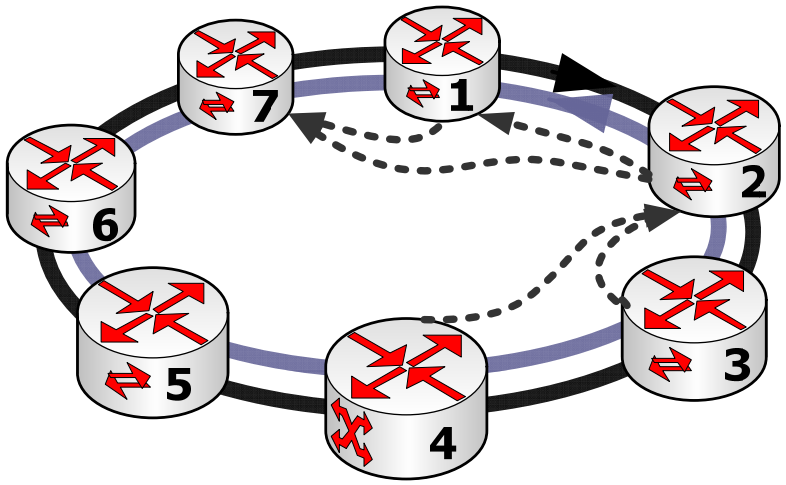

Figure 3.5 Destination stripping and spatial reuse illustrated on the inner ring.

RPR fairness is mainly based on ingress aggregation. This fairness is referred to as "Ring Ingress Aggregated with Spatial Reuse" (RIAS) fairness in an earlier article [35]. This definition follows the same methodology used in [41] for max-min flow control. The RIAS fairness definition, does not include the station weights in the

generalized formula while this is included in the IEEE 802.17 standard in the calculation of the estimated fair rate of a station. In addition, the RIAS definition assumes equal sharing of the bandwidth among flows originating from the same station, while the standard does not require that. In this section, a more general definition will be provided with the inclusion of the destination station weights along with the source station weights to provide a more comprehensive representation of fairness with respect to the IEEE 802.17 standard. This definition will be named as weighted RIAS (wRIAS).

Denote N as the total number of stations on a ringlet. Let the capacity of each link on the ringlet be C. Each Station s on the ringlet is given a weight w_s for providing the weighted fairness. On this ringlet, a flow vector is defined by $\boldsymbol{F}=\{f_{st}\}$, in which each flow from Station s to Station t is denoted by f_{st} which is also referred to as the path of the flow. For each flow a weight vector is defined by $\mathbf{P}=\{\rho_{st}\}$, in which the weight of each flow from Station s to Station t will be denoted by ρ_{st}. A fair rate vector is defined by $\mathbf{R}=\{r_{st}\}$ in which the fair rate of flow f_{st} is denoted by r_{st}. By using the above definitions, the total allocated rate on link n of the ringlet is given by Equation (3.1).

$$T_n = \sum_{\forall s,t:\ link\ n \in f_{st}} r_{st} \tag{3.1}$$

On this ringlet, the vector **R** is said to be feasible if the following conditions in Equations (3.2) and (3.3) are met.

$$r_{st} > 0 \quad \forall s,t : f_{st} \in \boldsymbol{F} \tag{3.2}$$

$$T_n \le C \quad \forall n \in \boldsymbol{N} : 0 < n \le N \tag{3.3}$$

The sum of all flows originating from Station s and passing through link n is

$$A_n(s) = \sum_{\forall t \in N:\ link\ n \in f_{st}} r_{st} \tag{3.4}$$

For a feasible vector **R**, the link *n* is a bottleneck link, $\mathbf{B_n}$(s,t), with respect to **R** for f_{st} crossing link *n* if the following conditions in Equations (3.5), (3.6), and (3.7) are met with respect to all flows $f_{s't'}$ crossing link *n*.

$$T_n = C \tag{3.5}$$

$$r_{s't'} \le r_{st} \quad \forall s', t' : s = s' \,\&\, t' \ne t \,\&\, link\ n \in f_{s't'} \tag{3.6}$$

$$A_n(s') \le A_n(s) \quad \forall s', t' : s' \ne s \,\&\, link\ n \in f_{s't'} \tag{3.7}$$

Note that if there are no other flows originated from any station other than Station *s* going through link *n*, $A_n(s')$ will be zero and Equation (3.7) will be satisfied by default.

The vector **R** is said to be "weighted" ingress aggregated fair with destination differentiation if it is feasible as defined in Equations (3.2) and (3.3) and if for each f_{st}, r_{st} cannot be increased while maintaining feasibility without decreasing the fair rate $r_{s't'}$ of some flow $f_{s't'}$ for which

$$\frac{r_{s't'}}{\rho_{s't'}} \le \frac{r_{st}}{\rho_{st}} \quad \forall s', t' : s' = s \,\&\, f_{s't'} \in F \tag{3.8}$$

$$\frac{A_n(s')}{w_{s'}} \le \frac{A_n(s)}{w_s}$$
$$\forall s', t', n, : s' \ne s \,\&\, f_{s't'} \in F \,\&$$
$$link(n) \in f_{s't'} \,\&\, link(n) \in f_{st} \tag{3.9}$$

Equation (3.8) ensures the fairness among the flows originating from the same station with destination differentiation, while Equation (3.9) ensures the fairness among ingress aggregated flows. The weights ρ and *w* are used to normalize the comparison and hence to achieve the weighted fairness for both destination flows and ingress aggregation, respectively.

For the scenario given in Figure 3.5, if Station *4* has two times more weight than Station *3*, it will get two times more bandwidth out of the ring than Station *3*. In this case,

if Station *3* increases its share, Equation (3.9) will not be satisfied. Destination differentiation can be provided at Station *2* if the destination weights (ρ_{21} and ρ_{27}) are adjusted so that the destination Station *1* has two times more weight than Station *7*. Then the Station *1* will receive two times more traffic from Station *2* as compared to Station *7* in order to satisfy Equation (3.8). Destination differentiation can be provided at Station *2* if the destination weights (ρ_{21} and ρ_{27}) are adjusted so that the destination Station *1* has two times more weight than Station *7*. Then the Station *1* will receive two times more traffic from Station *2* as compared to Station *7* in order to satisfy Equation (3.8).

3.2.2 Weighted Fairness Scenario

In this section, an example of a weighted fairness scenario, which demonstrates how the weights on an RPR ring are utilized, will be investigated. Note that in this scenario the destination weights ρ will be set to 1.

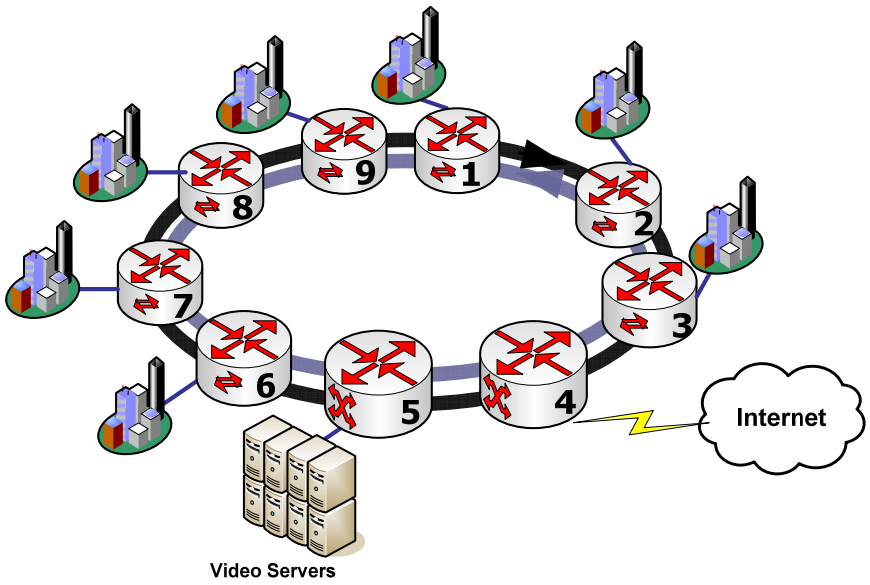

Figure 3.6 Weighted fairness scenario.

Figure 3.6 shows an example in which a service provider offers Internet and video service over its RPR network using an OC12 ring. The provider needs to make sure that there will be enough bandwidth on the ring to accommodate the video requests of the subscribers.

The video server is connected to Station *5* and the Internet connection is through Station *4* on the ring. Assume that the service provider is utilizing MPEG4 compression for a high definition video service where each connection is taking approximately 8Mbps of bandwidth 17 [42]. Also assume that in this scenario, a total of 50 different channels are being requested by the customers of the video service. This requires a total of 400Mbps of traffic to be originated from Station *5*. These video service customers are connected to Stations *6*, *7*, and *8* on the ring. At the same time, some other 200 customers with 1.5Mbps Internet connection services at Stations *6*, *7*, and *8* are downloading files through the Internet, generating a total amount of traffic of 300Mbps. For simplicity, other stations will not be included in the discussion and only the outer ringlet will be used in this example.

In the case of RIAS fairness, which does not account for weights, stations on the ring will share the ring bandwidth equally. This means that Station *4* and Station *5* will add an equal amount of traffic to the ring when there is congestion. This will be the case when there is a total of 400Mbps video and 300Mbps of Internet traffic being requested on an OC12 (~600Mbps net data throughput) ring. In this case, Station *4* will become the congestion tail and Station *5* will become the congestion head. Each of the Stations *4* and *5* will add approximately 300Mbps of traffic on to the ring. Therefore, the service provider will not be able to accommodate the requests for 50 different channels. In this

scenario, only 37 different channels can be distributed unless the service provider adjusts the parameters of the RPR network.

The issue can ideally be resolved by assigning weights to the stations on the ring. When there is a contention for resources, the weights will control the RPR network operation. The service provider can estimate the maximum bandwidth that will be expected from Station *5*. For the scenario being discussed, this is 400Mbps. Under normal conditions, Station *4* will be the next biggest contender for the ring bandwidth. Under the worst case, Station *4* should get the rest of the bandwidth, which is approximately 600-400=200Mbps of bandwidth. Since the ratio between these estimates is two, a weight of two can be assigned to Station *5*, while the weight of Station *4* will remain as one. This setting will ensure that customers will be able to enjoy watching 50 different programs simultaneously with the other 200 customers sharing the remaining 200Mbps of bandwidth on the outer ringlet.

3.2.3 Simulation Results

The scenario is simulated using the modified Simula RPR simulator [43] to allow per station weight adjustment. The simulation model is implemented in J-Sim [44] using Java. An OC12 ring which is composed of nine stations is created with 20km of distance between every two adjacent stations. Each station is configured as a dual-queue station with the aggressive fairness mode enabled. The size of the secondary transit queue (STQ) at each station is 512KB and the "lp_coef" [1] parameter of the RPR MAC is set to 16.

Figure 3.7 shows the total traffic sourced by Stations *4* and *5* to the outer ringlet starting at time 0.1s. As expected, the available bandwidth is being shared by Stations *4*

and *5* equally, which is around 300Mbps and the total amount of traffic sourced by all active stations (only *4* and *5* in this scenario) is around 600Mbps.

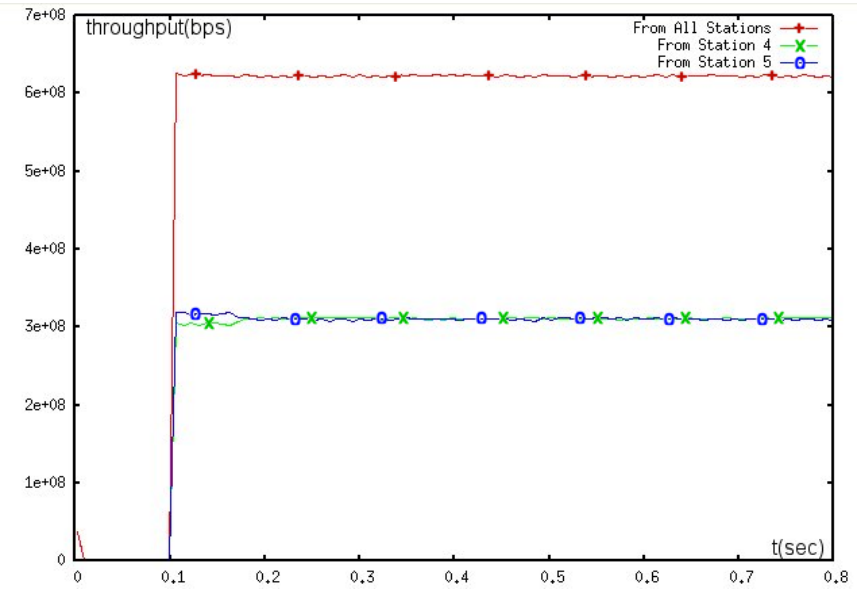

Figure 3.7 Throughput vs. time graph where Stations 4 and 5 have equal station weights.

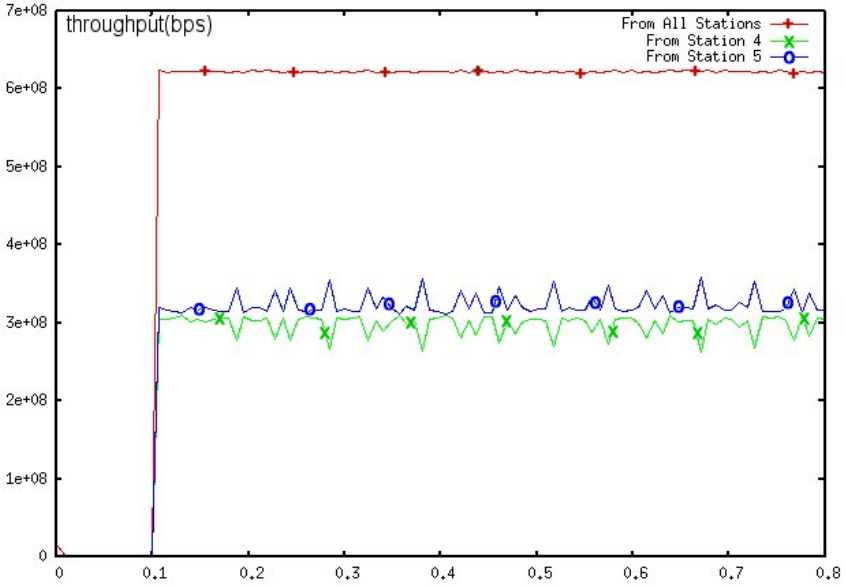

Figure 3.8 Throughput vs. time graph where Station 5 weight is set to 2.

Next, the weight of Station *5* is increased to two so that the station will take two times more of the fair bandwidth. Figure 3.8 shows the result for this scenario. However,

the desired behavior could not be observed. Another interesting observation from Figure 3.8 is that the throughput is oscillatory.

In order to test out the behavior further, another scenario is explored. In this scenario, the locations of the video server and the Internet connection are swapped so that Station *4* becomes the video server and Station *5* provides the Internet connection.

Figure 3.9 shows the result of this scenario. Interestingly, this scenario behaves as expected and the new video server (Station *4*) is able to acquire two times more bandwidth out of the ring than what Station *5* gets. In the next section, the cause of this response will be investigated and some suggestions will be provided to improve the behavior.

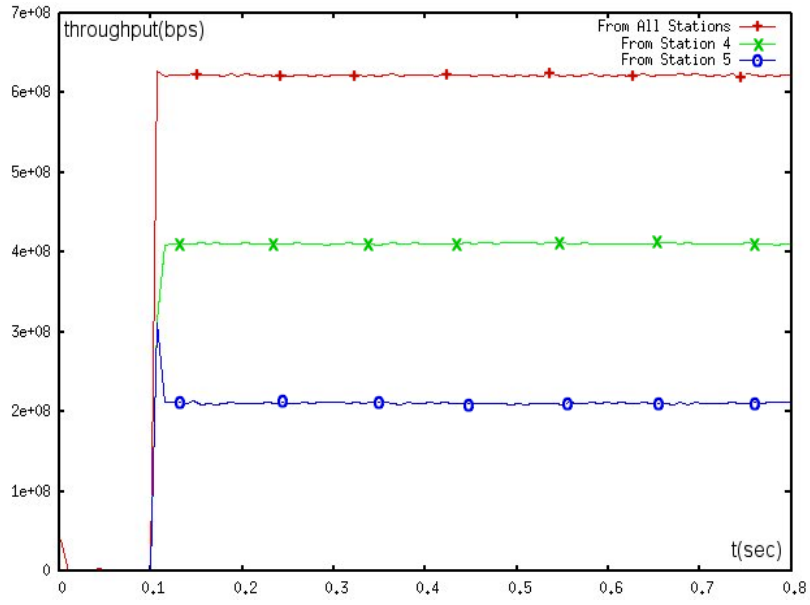

Figure 3.9 Throughput vs. time graph in which the stations of the video server and the Internet connection are swapped.

3.2.4 Analysis of Weighted Fairness Behavior

In this section, the behavior of the fairness algorithm will be investigated, upon which an improvement will be suggested.

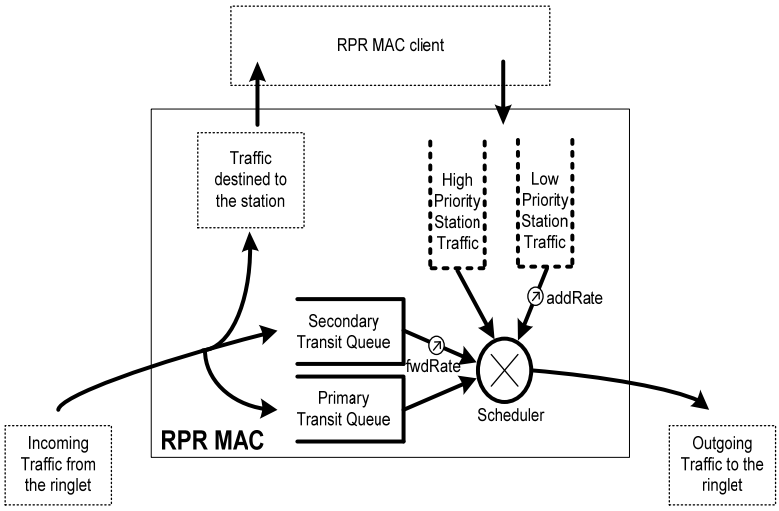

Figure 3.10 RPR MAC of a dual queue station.

Figure 3.10 shows a simplified RPR MAC of a dual queue station. The primary transit queue shown is dedicated to high priority ring traffic while the secondary transit queue is used for the remaining traffic in the dual-queue configuration.

For this scenario, the high priority traffic condition can be ignored because, the scenario did not include any high priority traffic. Then, the main scheduling decision is among packets from the STQ of the station and the low priority station traffic. This decision is called "addRateOk" in the IEEE 802.17 standard. If the "addRateOk" parameter is evaluated as true, then the low priority station traffic will be selected, otherwise the secondary transit queue will be selected. This decision is controlled by the "addRateOk" of RPR as shown in Equation (10) [1].

$$\begin{aligned} addRateOk = \; & (addRate < allowedRate) \;\&\& \\ & (nrXmitRate < unreservedRate) \;\&\& \\ & (STQ.empty() \;\|\| \\ & \quad (fwdRate > addRate) \;\&\& \\ & \qquad (STQ.depth() < stqHighTh))) \end{aligned} \tag{3.10}$$

The parameters "fwdRate" and "addRate" (also shown in Figure 3.10) are the rates of fairness eligible traffic (traffic that is regulated by the fairness algorithm) from

the secondary transit queue and the station, respectively. The "allowedRate" is the fair rate at which the station is allowed to add fairness eligible traffic. The "nrXmitRate" is the rate of traffic other than reserved high priority traffic on the ringlet. The "unreservedRate" is the difference between the link rate and the total reserved bandwidth (for high priority traffic) on the ringlet. The scheduler also monitors the STQ state ("STQ.empty" and "STQ.depth") and compares the occupancy of the STQ with a predefined threshold called "stqHighTh" for selecting which packet to transmit.

The first two parameters are not related in our example since Station *5* is the head of the congestion and there is no reserved traffic on the ring. Therefore, these expressions will always be evaluated as true in our scenario, and will be "don't care" for the expression. The third expression checks for the availability of a packet in the STQ. If there is a packet, it ensures the fair distribution of bandwidth unless the STQ occupancy has reached the high threshold level. The fair distribution in this case is equal bandwidth for both of the transit and station traffic. Thus, this fair distribution of bandwidth is the culprit. When the station is assigned to a higher weight, it is supposed to get a weighted share out of the ring. In order to accomplish the desired behavior, the addRate needs to be normalized so that the station can schedule packets in a weighted manner. Equation (3.11) shows the improved equation to resolve the unexpected behavior, which includes the "**localWeight**" factor. This factor will allow the station to add "**localWeight**" number of bytes on to the ringlet for each byte forwarded from the STQ.

$$
\begin{aligned}
addRateOk = \; & (addRate < allowedRate) \;\&\& \\
& (nrXmitRate < unreservedRate) \;\&\& \\
& (STQ.empty \;\|\| \\
& \quad (fwdRate * \mathbf{localWeight} > addRate) \;\&\& \\
& \qquad (STQ.depth < stqHighTh)))
\end{aligned}
\tag{3.11}
$$

The current calculation in the IEEE 802.17 standard shown in Equation (3.10) will not allow the current station to transmit enough bytes when the "fwdRate" and "addRate" parameters are compared even if the station is given a higher weight. This will cause the station to slow down when the station with the higher weight is the congested station. This behavior is not observed as shown in Figure 3.9 when the station with the higher weight is an upstream station. The reason is that the "allowedRate" (estimation of the fair rate) in 802.17 already includes the station weights and in this case the "fwdRate" and "addRate" comparison will not be considered to be true if the station is not congested. By adding the "localWeight" factor as in Equation (3.11), the station with the higher weight will have a better chance to transmit add traffic as compared to the transit traffic in accordance with its assigned weight. After changing the equation in the simulation model as in Equation (3.11), the scenario has been tested once more and the following results shown in Figure 3.11 have been obtained as opposed to the results shown in Figure 3.8.

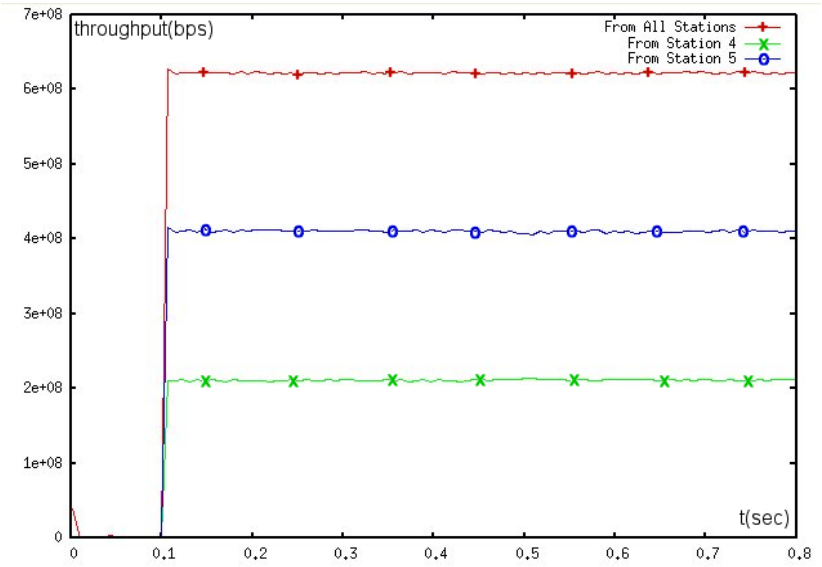

Figure 3.11 Throughput vs. time graph where Station *5* weight is set to 2 with the updated addRateOk calculation.

This time the results are in line with the expected behavior and the bandwidth is shared according to the weights of the stations. In other words, Station *5* is getting two times more bandwidth than Station *4* on the ringlet. Specifically, Station *5* is adding 400Mbps and Station *4* is adding 200Mbps of traffic to the outer ringlet.

The addition of a new factor in the calculation of addRateOk parameter requires an additional multiplication operation in the scheduler. To simplify the calculation, one can require the weight to be power of two so that a simple shift operation can replace the complicated multiplication circuitry. Another option is to add a new parameter called "weightedFwdRate", and per each byte transmitted from STQ, increment the "weightedFwdRate" with the weight of the station. A network operator may also follow other practices to avoid encountering the scenario discussed above. One of them, as shown in Figure 3.9, is to make sure that a station with a larger weight does not become a head of the congestion domain. Also, another desirable approach is to distribute the high throughput servers evenly around the ring when possible, because this will allow efficient use of both ringlets and will decrease the contention on the ring.

The last item to note is the oscillations observed in Figure 3.8. This is mainly due to the feedback control mechanism of RPR in the aggressive mode of operation. Once the STQ reaches a certain threshold (in the aggressive mode of operation), a station is considered to be congested. At this point, the station starts transmitting a message with its own normalized addRate to the upstream stations. Once an upstream station receives this message, it will adjust its transmit rate to the fair rate (addRate) of the station that transmitted the congested message. In this case, the video server transmits one half of its own addRate to the upstream stations. When the upstream station receives the

notification, it slows down to this rate. However, there are already packets waiting in the STQ of the congested station and the scheduler is transmitting those packets. Once the station lets some of those packets in the STQ go, the station is no longer congested and stops transmitting its own normalized rate, which in turn lets Station *4* increase its share on the ring. This mechanism creates an oscillatory behavior in this specific case which can be smoothed out by increasing the available STQ size, and this will result in equal sharing of the ring bandwidth (which is not desired in this scenario). On the other hand, if the STQ size is decreased, there will be more oscillations while the ratio of traffic added by each station will approach the ratio of station weights.

3.2.5 Weighted Fairness under Instability

It was shown in [45] that the RPR algorithm can suffer from oscillations under some special scenarios where the congested station has little traffic. It is quite clear that the network utilization will go down as a result of these oscillations. In this section, how weighted fairness can be utilized to alleviate underutilization of the network will be investigated.

The earlier weighted fairness scenario which was shown in Figure 3.6 is modified to create the oscillatory behavior. Note that the updated weighted fairness algorithm is used in the simulation.

In this scenario, Station 4 has 400Mbps, Station *5* has 300Mbps, and Station *6* has 20Mbps of traffic, all of which are destined to station *7*. The service provider still wants to make sure that Station *4* will get 400Mbps when needed to support 50 different channels.

The difference from the previous scenario shown in Section 3.2.2 is that the only destination is Station *7* and a new traffic source, Station *6*, is added. Note that all the stations have a weight of one. The oscillations are observed as a result of having Station *6* adding very small amount of traffic while being the congested station at the same time. As Station *6* gets congested, it advertises its current add rate. This slows down Station *4* and Station *5* more than they should periodically, hence resulting in the oscillatory behavior as shown in Figure 3.12.

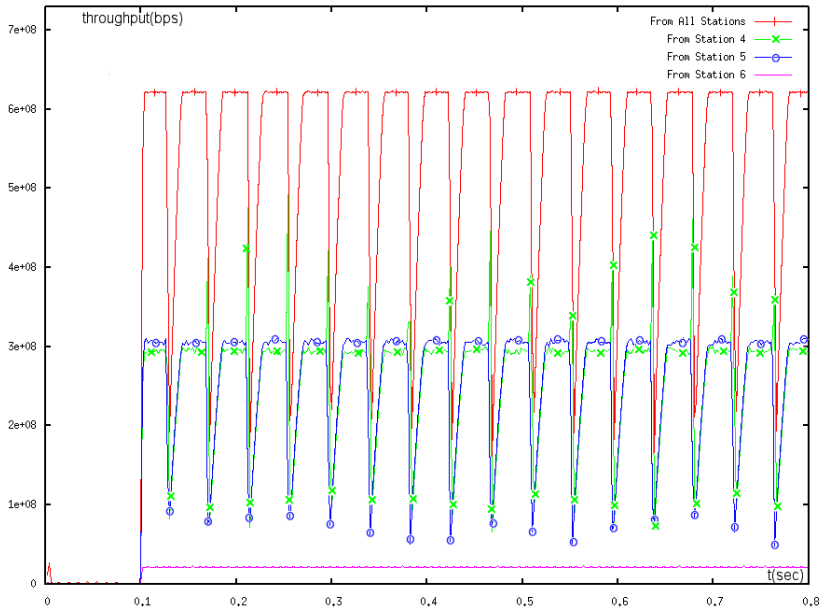

Figure 3.12 Throughput vs. time of the scenario where the weight of stations are set to 1.

The next scenario has the same traffic pattern; however, the weights of the stations are different. The weight of Station *4* is 10, the weight of Station *5* is 20, and the weight of Station *6* is 1. In this case, the stations share the bandwidth as desired and the oscillations are gone as shown in Figure 3.13.

Note that the period of the congestion interval depends on the amount of traffic added to the ring by Station *6* when other parameters such as ring size and buffer

thresholds remain the same [38]. When that traffic decreases, the congestion interval will increase. Under this condition, even though some oscillations might still be observed, the impact on the total network utilization will be minimal. If traffic added by Station *6* increases, the fairness algorithm will function better as the congested station (Station *6*) will have more traffic to advertise.

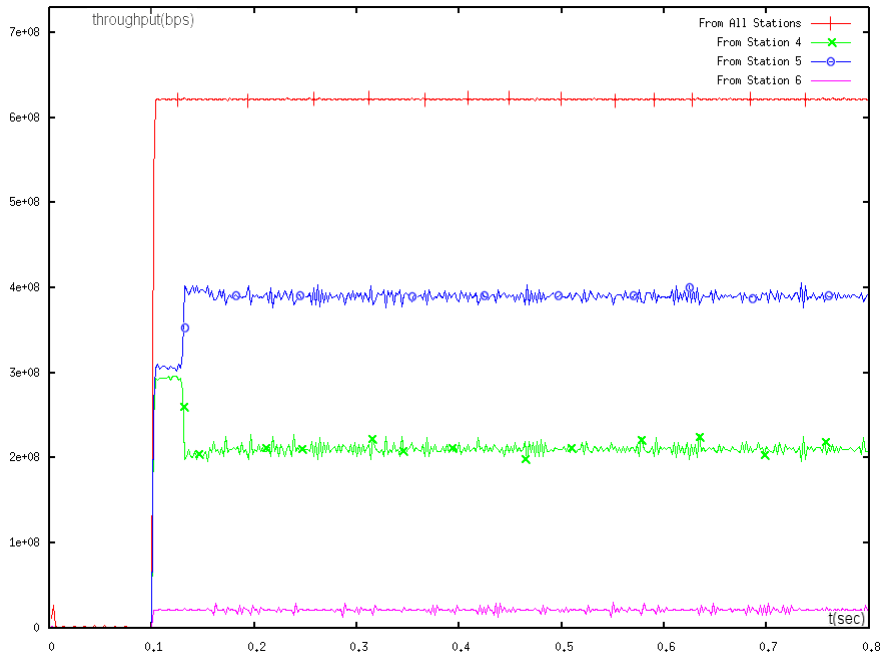

Figure 3.13 Throughput vs. time of the scenario where the weight of Station *5* is set to 20 while the weight of Station *4* is set to 10.

The adjustment of weights should not be confused with static bandwidth assignments. The reason is that the weights will only be active if there is traffic from the station with higher weight and there is some congestion down the path. Otherwise, the stations which are assigned with smaller weights will utilize the unused bandwidth. In addition, the adjustment of weights is well suited to the current network architectures where the upload limit for the stations in the network is generally much less than the download limit.

Note that the behavior of the fairness algorithm is tightly coupled with the secondary transit buffer thresholds, round trip time of the network, and the amount of smoothing of the instantaneous measurements [38]. By changing these parameters intelligently, the network behavior can be optimized further.

This section has discussed and explained the use of weighted fairness in an RPR network. It has extended the definition of ring ingress aggregated fairness by incorporating weights into the formulation. Performance evaluations by using the latest version of the IEEE 802.17 RPR standard have demonstrated how the bandwidth is shared by using different weights. In particular, a pitfall was identified and improvements are suggested to circumvent that pitfall as substantiated by the simulation results. In addition, it is shown that by adjusting various parameters already available in the fairness algorithm of IEEE 802.17, one can eradicate the oscillatory behavior under certain scenarios.

3.3 Multi-choke Point Detection and Virtual Destination Queuing for RPR

The SRP algorithm is focused on a single point of congestion, which sometimes may result in lower utilization of the network. Obviously, it is possible to increase the utilization by utilizing virtual destination queuing and distributing the congestion status of each station to every other station. A scheduling policy will then utilize the congestion status of the downstream stations as well as queue status. The underlying mechanism to implement this congestion status distribution was added to the RPR standard. This mechanism was initially discussed in the RPR Workgroup presentation [30]. This proposal also suggested multi-choke implementation to decrease the scheduling complexity of the virtual destination queuing algorithm. In addition, the text in Appendix

J of IEEE 802.17 RPR Standard [1] was initially provided to introduce the basic mechanism to implement a MAC client with virtual destination queuing.

3.3.1 Usage Packet Handling

Each station generates usage messages to distribute the total usage value of that station. When a station is not congested, a special message with "not congested" information will be generated. A usage packet is removed from the ring if the station, which generated the usage message, receives its own usage message back.

3.3.2 Virtual Destination Queues and Scheduling

To support full virtual destination queuing, a station is required to incorporate as many queues as the number of stations on the ring. A station will update the appropriate choke point information when it receives the corresponding usage packet from a station. Stations limit the amount of insertion traffic sent through the choke points.

Supporting a large number of stations in MAC is not efficient. Instead, it is possible to pass the choke point information to the MAC client and the MAC client can handle the scheduling of virtual destination queues (VDQs). Usage values and allowed usages are decayed/incremented similar to SRP-fa [16].

Instead of supporting full virtual destination queueing, a MAC client may choose to keep track of a number of congestion points less than the number of stations on the ring. This will decrease the number of queues that the MAC client needs to implement. Therefore, the number of choke points supported will determine the tradeoff between implementation complexity and the achievable network utilization.

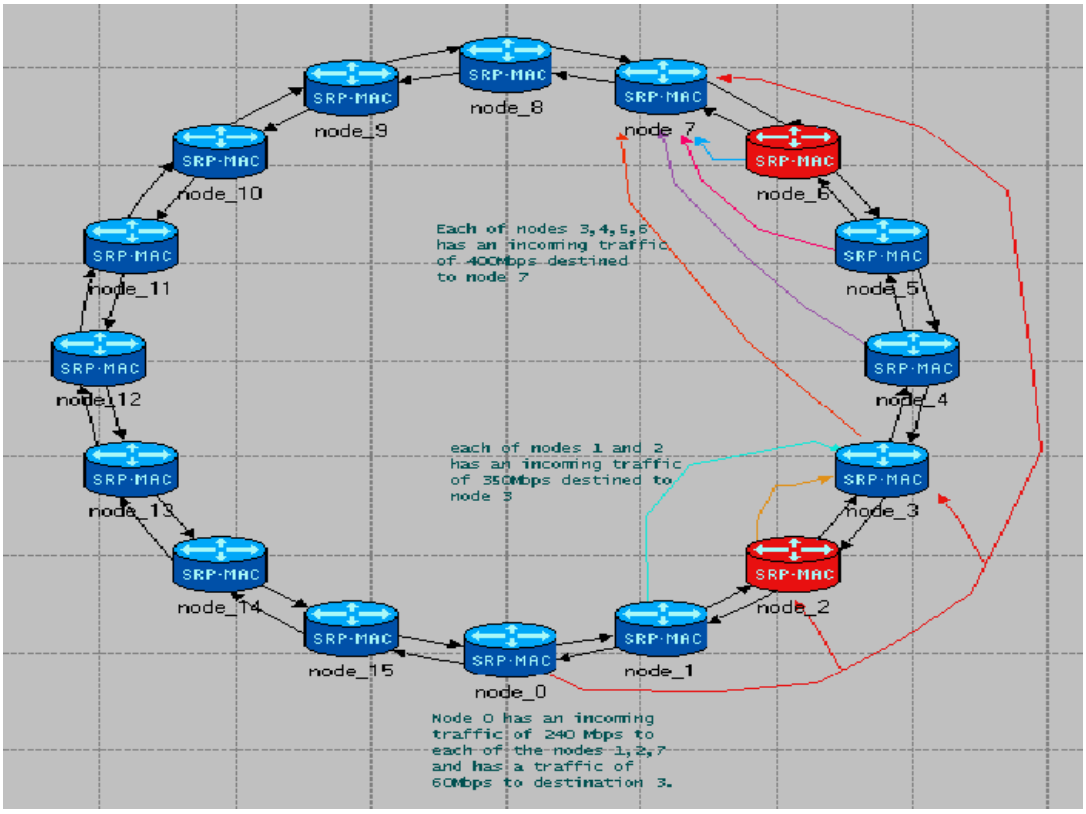

Figure 3.14 Congestion scenario.

Figure 3.14 shows an example scenario which exhibits two congestion domains. The first one is at Station *2* and the other one is at Station *6* while Station *0* is sourcing traffic for both congestion domains. In order to maximize bandwidth utilization, Station *0* needs to know the status of 2 stations, specifically Station *2* and Station *6*. In SRP, a station can only keep track of single congestion point. The amount of traffic which can be sourced by Station *0* is limited by the downstream congestion point. In this scenario, this is the congestion status of Station 6 which has the smallest usage value. If the multi choke algorithm is supported a station will be aware of multiple congestion domains because Station 0 will be receiving usage values u2 and u6 from Station *2* and Station *6*, respectively as shown in Figure 3.15. The first congestion domain comprises the Stations *1* and *2*. The second congestion domain comprises the Stations between Stations

3 and *6* (inclusive). Finally, the third congestion domain comprises the stations beyond Station *6*. Station *0* should obey the following constraints while scheduling its virtual destination queues:

1. Up to the line rate for traffic destined to Station *1* and Station *2*.

2. Virtual destination queues for Stations *3*, *4*, *5*, and *6* can be scheduled as long as the total usage beyond VDQ2 does not exceed u2.

3. Virtual destination queues for stations beyond Station *6* can be scheduled as long as the total usage beyond VDQ2 does not exceed u2 and the total usage beyond VDQ6 does not exceed u6.

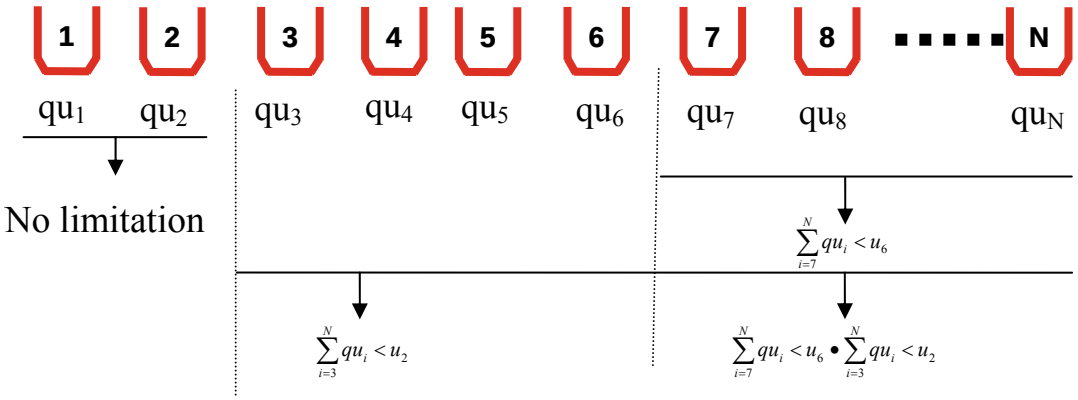

Figure 3.15 The scheduling constraints at Station 6.

Figure 3.16 shows the network utilization for an implementation that keeps track of four congested downstream stations. For the considered scenario, the same utilization can be reached even with a single choke point. This idea is utilized in IEEE 802.17 RPR and the single choke point is tracked by the MAC. This facilitates better utilization of the ring in a wide range of traffic scenarios over SRP.

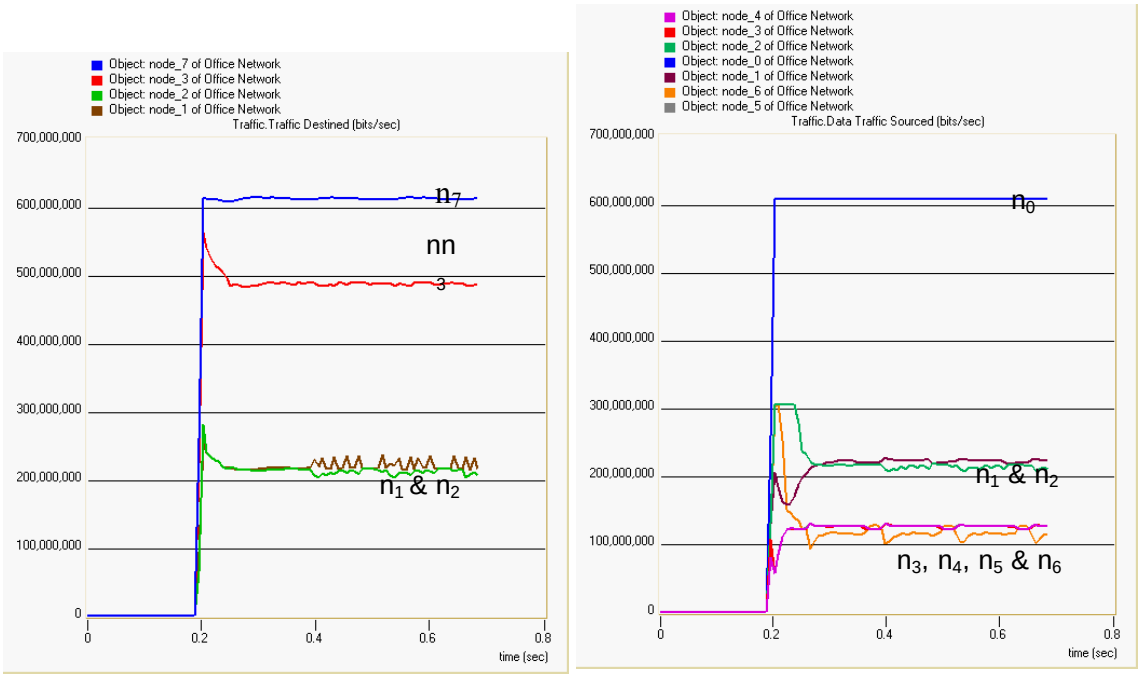

Figure 3.16 VDQ, max choke point is set to 4.

3.3.3 MAC Client Implementation of the Virtual Destination Queuing

The RPR standard defines a set of primitives at the MAC service access point (MSAP). The number of queues and the queue managers at the MAC client are a matter of choice. The simplest MAC client can have one queue for each traffic class. The STOP_LO, STOP_MED, STOP_HI signals will indicate the traffic class that cannot be sent. If the MAC client sends a packet of a stopped traffic class, the MAC policing functionality will not allow any more packets to be sent until that traffic class is allowed.

The MAC client can decide to send a medium priority packet as an excess bandwidth packet, in which case MAC will treat that packet as a low priority packet and the status of STOP_LO signal will be important. This means that a MAC client is allowed to send a medium priority packet even when there is STOP_MED signal, provided that STOP_LO is not asserted. When a medium priority packet must be treated as low

priority, MAC will mark the service class field in the header as out-of-profile [1], and the packet will consume rate shaping and fairness credits of the low priority traffic class in the MAC. It is possible to starve low priority traffic by sending excess medium traffic in place of low priority traffic. The MAC client should choose how much it could schedule excess medium priority traffic to starve or not to starve low priority traffic. The client can follow the queue selection policy shown in Table 3.1.

Table 3.1 Mac Client Queue Selection Policy

Stop_L	Stop_M	Stop_H	Which queue to select
0	0	0	If there is a packet in high class traffic queue, schedule high class If there is a packet in medium class traffic queue, schedule medium class If there is a packet in low class traffic queue, schedule low class
0	0	1	If there is a packet in medium class traffic queue, schedule medium class If there is a packet in low class traffic queue, schedule low class
0	1	0	If there is a packet in high class traffic queue, schedule high class If there is a packet in medium class traffic queue, schedule medium class (will be treated as low priority) If there is a packet in low class traffic queue, schedule low class
0	1	1	If there is a packet in medium class traffic queue, schedule medium class (will be treated as low priority) If there is a packet in low class traffic queue, schedule low class
1	0	0	If there is a packet in high class traffic queue, schedule high class If there is a packet in medium class traffic queue, schedule medium class
1	0	1	If there is a packet in medium class traffic queue, schedule medium class
1	1	0	If there is a packet in high class traffic queue, schedule high class
1	1	1	Stop scheduling any more packets

The MAC client may, as an option, implement a more sophisticated queuing scheme to avoid head-of-line blocking and to utilize more bandwidth. This can be accomplished through the use of network congestion information transmitted by the stations on the network and the collected topology information. For this purpose, the MAC client can implement virtual output queues for each destination on the ring for low and/or medium priority.

The MAC client is allowed to send a packet from a virtual output queue for low priority or excess medium priority queues if it can satisfy the necessary condition for each congestion point before the selected destination. The total usage beyond the congestion point should be less than the congested station's fairness value.

At any time there can be more than one virtual output queue which satisfies the condition. In this case, a round robin approach can be chosen to simplify the solution. However, a better approach will be using deficit round robin, which will avoid possible unfairness among virtual output queues. The algorithm implemented in the OPNET simulator to verify the idea is summarized as follows:

- Check the availability of a packet in a round robin fashion.
- A queue will be allowed to send if it has enough tokens. If a queue is not allowed to send, check the next queue.
- When a usage message is received from congested station, set the allowed usage of a queue to be the received value.
- Increase the allowed usage for choke points periodically.

The calculations of queue add rates and allowed rates beyond a congested point are also important factors to increase utilization and obtain a stable behavior. An acceptable approach is to choose a similar algorithm to update and increment these values

as in RPR MAC client for “allowed_rate” and “add_rate” for each virtual output queue. In addition, one should low-pass filter the value of per queue add rates to smooth out instantaneous variations. Once the MAC client receives a fairness message from MAC about a congested station, it will update the allowed_rate of that destination, which represents the total allowed_rate beyond that station. That value will then be incremented periodically as long as the MAC client does not receive another fairness message for that station. In essence, the MAC client implements a copy of the MAC fairness algorithm for each destination.

Depending on the client’s behavior, the assertion of STOP signals will vary in the MAC. For virtual destination queuing, ideally STOP signals will never be asserted other than rate shaping purposes. If a client misbehaves, MAC policing functions will prevent the client from abusing the ring.

3.3.4 Usage Messaging

There are two possible implementations of RPR fairness algorithm (RPR-fa). Basic RPR-fa is implemented completely in the MAC and does not have knowledge of the ring topology. Multi-choke RPR-fa is an enhancement to Basic RPR-fa that utilizes topology information along with per-destination transmit queuing to increase ring utilization.

There are two types of usage messages. The first type is store and forward basic usage messages. This type of usage messages are generated at every usage generation interval. The second type is ring wide distributed usage message. The second type of message is generated only when a station gets congested. These messages are not generated more often than 10 times the generation interval of basic usage messages. The

second type of usage messages are used to distribute every station's usage information all around the ring. These messages can be utilized by multi-choke capable RPR-fa stations.

If a station experiences congestion, it will advertise the value of its transmit usage counter to upstream stations via the opposite ring. The usage counter is run through a low pass filter function and normalized by the station's weight. The low-pass filter stabilizes the feedback and the division by weight normalizes the transmitted value to a weight of 1.0. When they receive an advertised usage value, upstream stations will adjust their transmit rates so as not to exceed the advertised value (adjusted by their weights). Stations also propagate the received advertised value to their immediate upstream neighbor. Stations receiving advertised values which are also congested propagate the minimum of their normalized low-pass filtered transmit usage and the received usage.

Multi-choke RPR-fa is an enhancement to RPR-fa that deals with the case where a station wants to send traffic to a destination that is closer than a congested link. As an example, consider the case where Station 1 wants to send traffic to Station 2, and the link between Stations 2 and 3 is congested. Basic RPR-fa will limit Station 1's traffic even though the congestion point is beyond the destination. Multi-Choke RPR-fa will allow Station 1 to send as much traffic as it wants to Station 2, and will only limit traffic to stations beyond the congested link.

If a Station gets congested, a second type usage message will also be generated. This usage message will traverse the ring without any modification (except the TTL field) and will be removed from the ring by the source station. In Multi-choke RPR-fa, each station will track advertised usage values for *n* congested station, where *n* is adjustable from 1 to half the number of stations on the ring. A station is allowed to send

unlimited traffic to any station between itself and the first congested station (choke point). It can send traffic to station between the first and second choke point based on the first choke point's advertised usage value. In general, a station can send traffic to a particular destination if it has satisfied the usage conditions for all choke points between itself and the destination.

Congestion is detected when the depth of the low priority transit buffer reaches a congestion threshold. The first type of usage messages, which are generated periodically, also act as keep-alives messages to inform the upstream station that a valid data link exists.

3.4 Transit Buffer Requirements for High Priority traffic

Dual transit buffers at each station will provide a way to differentiate high priority (HP) traffic from the low priority traffic on a ring. There are, however, some limitations on what can be guaranteed under some certain scenarios. Therefore, it is important to identify the cases and design the network accordingly.

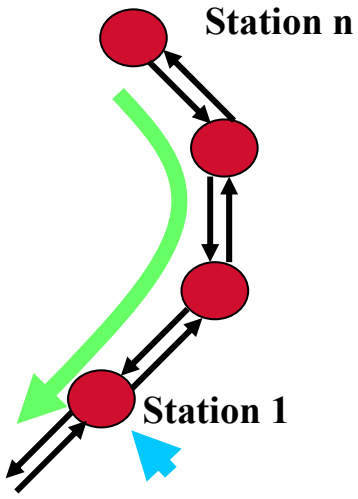

Figure 3.17 High priority traffic being injected at Station 1.

The highest utilization on the transit buffer will be attained when the furthest station is sourcing traffic to the ring at the line rate and the end station suddenly starts to inject high priority traffic. The scenario is shown in Figure 3.17. The farthest station (*n*-1 station away) generates traffic at the line rate through Station *1*. High priority traffic starts suddenly. In this scenario, as soon as the high priority traffic starts, the low priority transit buffer at Station *1* will start to fill up. This will trigger a slowdown message to upstream stations with a usage value of zero. When the end station (Station *n*) receives the slowdown message, it will stop sourcing the low priority traffic. If the transit buffer at Station 1 gets filled up during that time period, priority inversion will take place and high priority traffic will not be allowed any more into the ring till the buffer utilization goes below a threshold. Priority inversion is allowed on a ring because one of the goals of a resilient ring is to provide a lossless medium. Admittedly, priority inversion will cause a service quality degradation which is not desired. To prevent priority inversion, each station should have sufficient amount of buffer at each station.

To calculate the worst case buffer requirements, one needs to find out the time that will elapse from the instant Station *1* generates a slowdown message until it starts to observe the effect of its message which is the decreased amount of low priority traffic from the upstream stations.

The time it takes for the message to propagate back to the station has two components as shown in Figure 3.18. The first component which is the propagation time (t_p) is the total distance between two stations. The other component will be the time that is lost at each station. Since the worst case is being investigated, the usage generation interval will be used as the response time at each station.

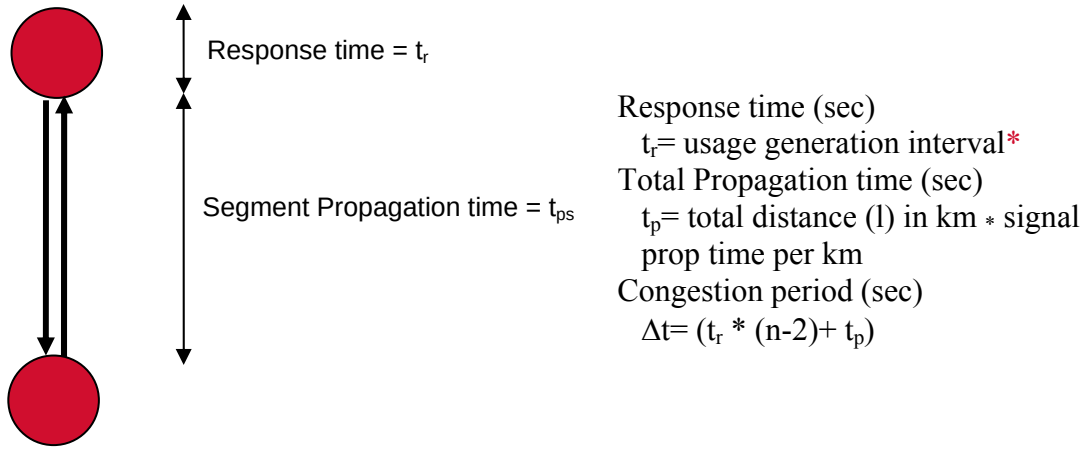

Figure 3.18 Usage message propagation time.

The amount of traffic that needs to be buffered during the congestion period is the amount of traffic sourced during that period and the traffic queued at each station.

$$\begin{aligned} BSz_M &= (\Delta t + t_p)\cdot LR + (n-2)\cdot LTH \\ &= ((n-1)t_r + 2t_p)\cdot LR + (n-2)\cdot LTH \end{aligned} \tag{3.12}$$

Equation (3.12) shows that the required buffer size is a function of the circumference of the ring, the number of stations on the ring, the minimum buffer threshold (LTH), the usage generation interval and the line rate of the ring.

Consider a 16-station, 300km ring with LTH=40kB, LR=10Gbit and usage interval being 10μs. As long as there are not any wraps, only half of the ring will be actively used. So *n* can be assumed to be 8.

$$\begin{aligned} BSz_M &= ((8-1)\times 10\times 10^{-6} + 2\times 300/2\times 5\times 10^{-6})\times 10\times 10^{9} + 6\cdot 40\times 1024\times 8 \\ &= 17.6x10^{6}\, bit \end{aligned} \tag{3.13}$$

Therefore, for this scenario the amount of buffer required will be around 2.1MB. The actual amount of allowed high priority traffic can be factored in to the equation to decrease the buffer requirements.

3.5 Worst-case Jitter Analysis of RPR

This section reviews the performance in aggressive mode of operation. Some of the simulations discussed in this section was utilized in the RPR Workgroup Presentation [31]. Part of this information was used to compare the aggressive and conservative modes of operation.

One can calculate the worst-case jitter for HP traffic on a ring network with 2-transit buffers at each station if the following conditions are met.

- The low priority transit buffer size is correctly chosen for the ring size so that the low priority transit buffer will never reach the high threshold. Correct sizing of low priority transit buffers ensures that priority inversion will not happen at a station.

- The total high priority traffic being sourced on to the ring does not exceed the link rate.

- HP traffic is shaped before being sourced into the ring at each station to prevent bursty traffic.

Under the limitation of the above conditions, one can look at the case when a station wants to insert an HP packet into the network to calculate the worst-case jitter. The packet will be queued until the high priority buffer gets emptied and the station completes the previous packet that is being transmitted. All stations are transmitting HP packets simultaneously adjusted by the link propagation delay. The *N*th station could also have a packet already in transit at this time. Therefore, on a ring of 2*N* stations, the best possible delay-jitter is (*N*+1)*MTU because, half the ring has *N* stations and there could be a packet in transit at this time at the *N*th station.

In Figure 3.19, there is an OC-192 ring of 16 stations. Stations from node_0 to node_6 are sourcing traffic to node_7. Each of the stations has 1100Mbps of low priority

traffic and 400Mbps of high priority traffic. In the example shown in Figure 3.19, the MTU is set to be 534B. When an HP packet is sourced at node_7, it will wait for the completion of the packet that is already being transmitted. After that, at each station, there can be another packet being transmitted. Therefore, the packets may be delayed at most by seven times.

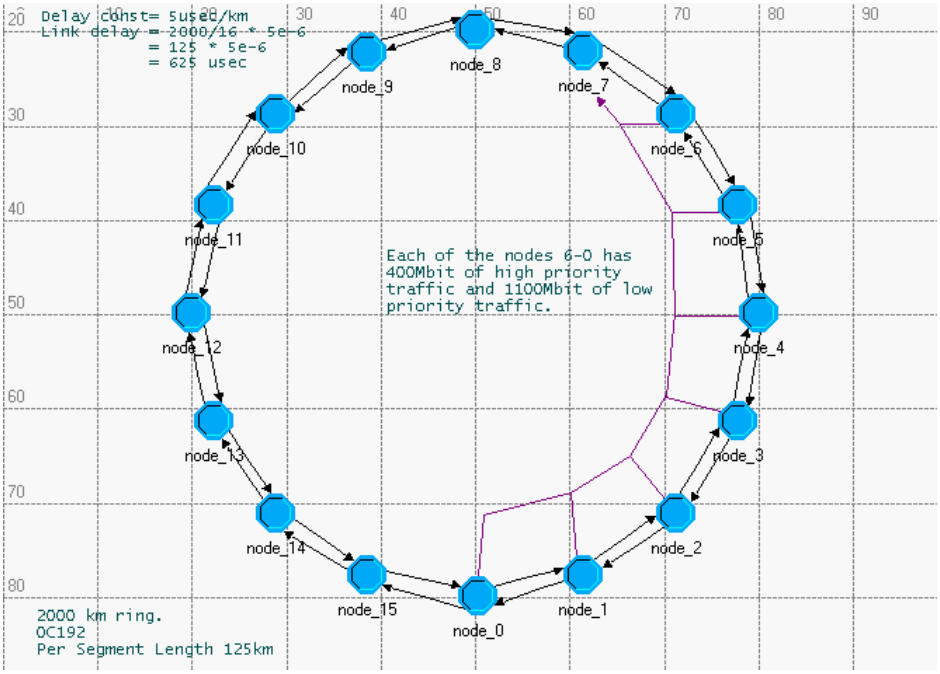

Figure 3.19 Example scenario for jitter measurement.

The important point is that one has to shape the high priority traffic and should not allow bursts of high priority traffic into the ring from a single station. The station buffers should also be adjusted accordingly to prevent priority inversion. These two important points have been taken into account and implemented in IEEE 802.17 RPR to provide deterministic jitter performance for the high priority traffic. As shown in delay distribution in Figure 3.20, the observed worst-case jitter is 2.96µsec (4.38096-4.37800) which is very close to the estimated value of 2.99µsec (7 MTU time).

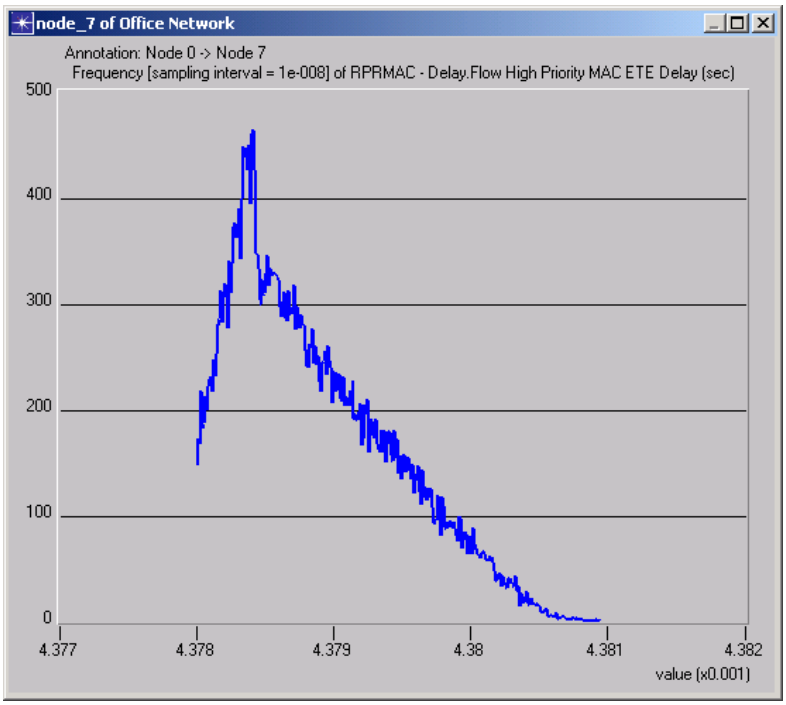

Figure 3.20 Delay distribution observed by HP packets sourced by node_0.

3.6 Limiting Forward Rate for Uncommitted Traffic in RPR

It is not possible, in some cases, to provide enough buffering which is required to prevent priority inversion. When this is the case, one option is to limit the total amount of low priority traffic allowed into the ring. This can be achieved by not serving the low priority packets when the following condition in Equation (3.14) is met:

$$my\_usage + fwd\_rate < unreserved\_bandwidth \qquad (3.14)$$

This means that MAC will not schedule low priority packets from transit and transmit buffers when the "limit check expression" is true. This improvement over the SRP algorithm is incorporated into the IEEE 802.17 RPR standard. The overhead of in-band control messages should be taken into account to make precise adjustments in reserving bandwidth.

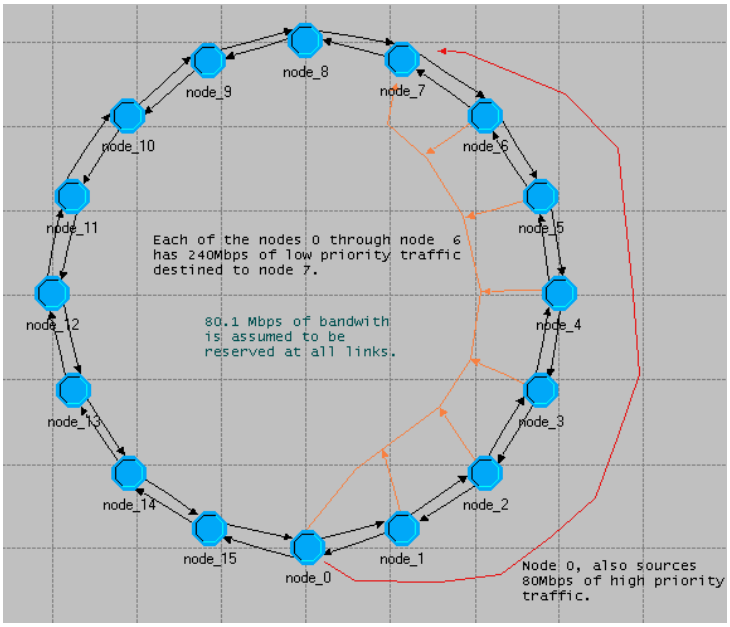

Figure 3.21 Hub scenario.

Figure 3.21 shows the hub scenario where the modified SRP algorithm is running at each station, with "unreserved_bandwidth" parameter set to 542Mpbs. Stations node_0 to node_6 have 240Mpbs of low priority traffic, while node 0 has 80Mbps of high priority traffic.

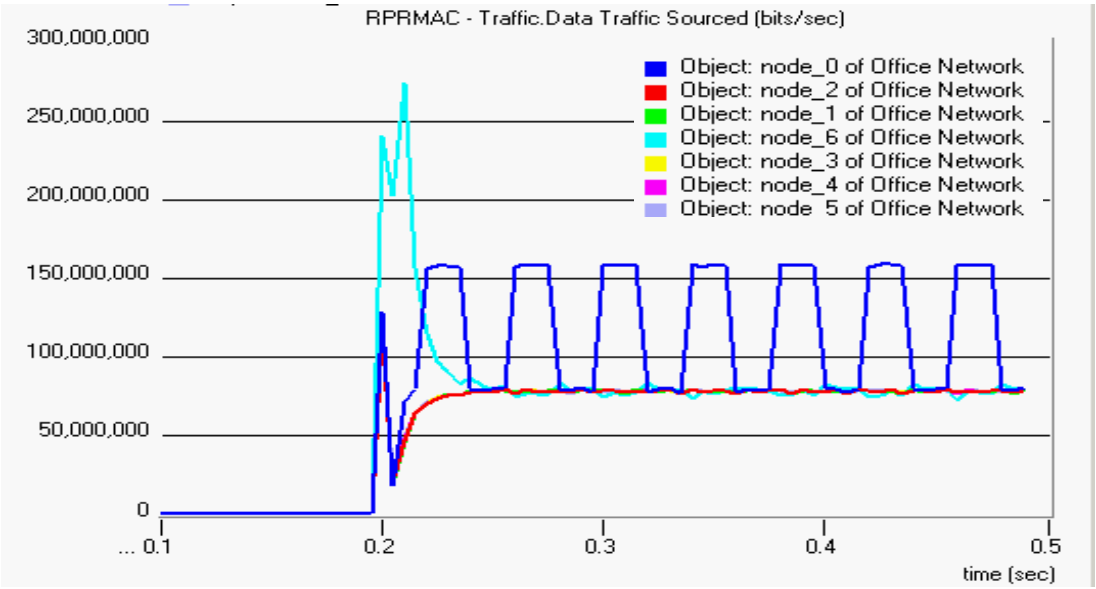

Figure 3.22 Data traffic sourced by the stations.

Figure 3.22 shows the total data traffic sourced by the stations. Station node_0 has the additional high priority traffic which has an on and off pattern which is shown in Figure 3.23.

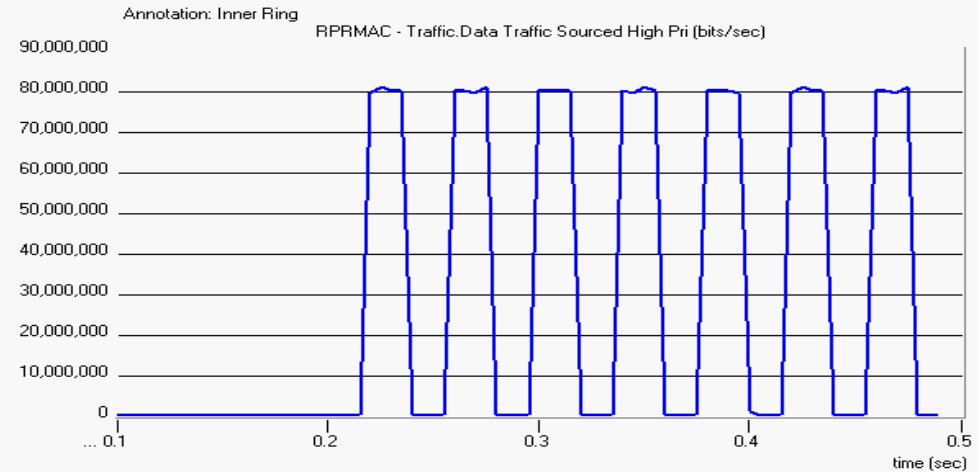

Figure 3.23 High priority data traffic sourced by node_0.

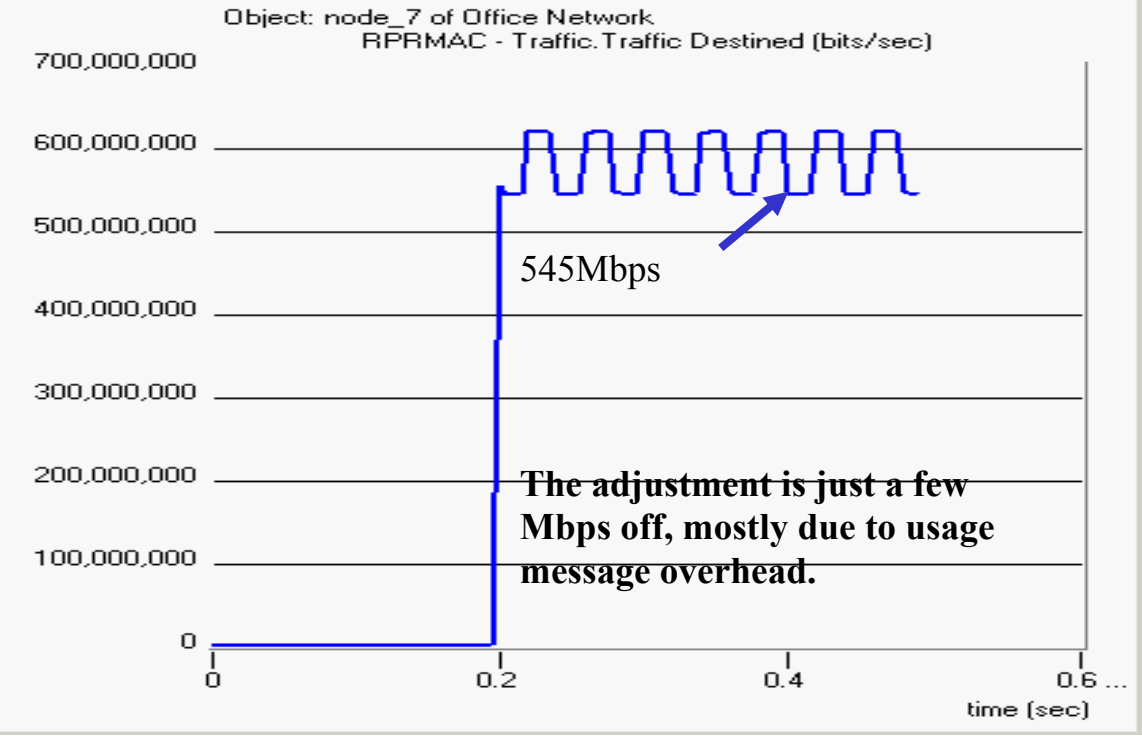

Figure 3.24 Total data traffic received at node_7.

Since the ring is limited to 542 Mpbs for low priority traffic, each of the stations ends up sourcing 77.5 Mbps of low priority traffic. In addition, the high priority traffic can easily be inserted into the ring without affecting low priority traffic.

Figure 3.24 shows the traffic received at Station node_7. It is observed that when there is no high priority traffic, the maximum bandwidth that can be utilized on the ring is limited by 545 Mbps, which is higher than the adjusted value of 542Mbps due to the additional overhead of usage messages.

The same scenario is run without bandwidth reservation and the results are shown in Figure 3.25 Even though the total bandwidth is being utilized all the time, some oscillations are observed due to bandwidth reclamation.

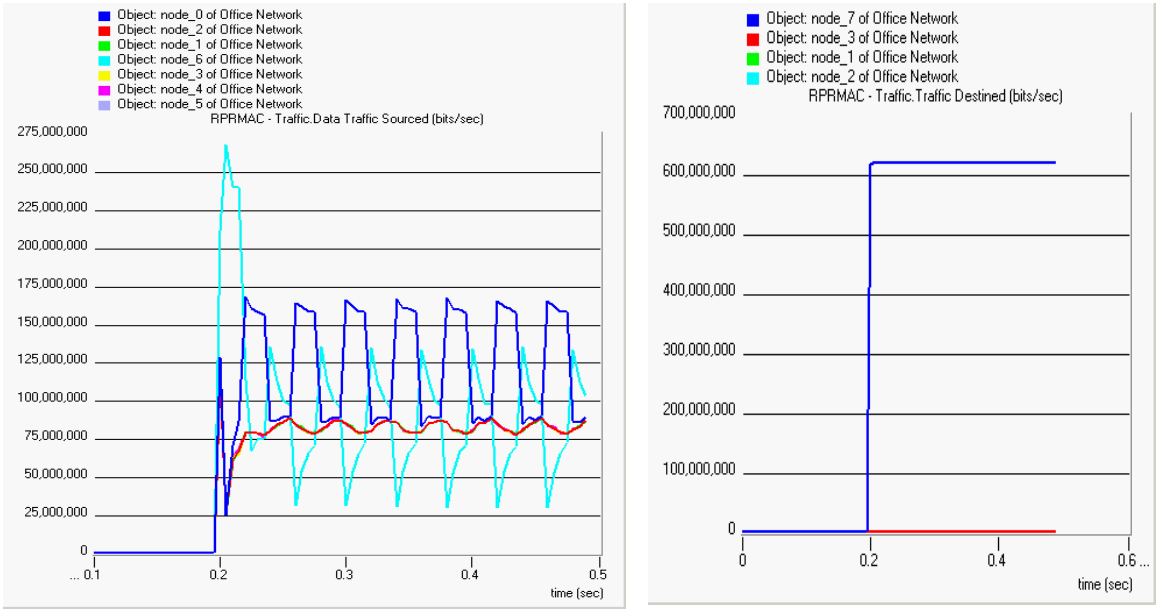

Figure 3.25 Hub scenario with no bandwidth reservation.

3.7 Sizing of Secondary Transit Queue

The control mechanism of RPR has been very well studied and new algorithms have been suggested [35,39,40]. These algorithms do require modification from the current standard. Appendix G of the RPR standard [1] provides implementation guidelines. It

does not specify all the requirements for STQ sizing. Specifically, the guidelines provided in the standard are not satisfactory for the underflow case. The following discussion is based on [46].

While the transit queue sizing has been investigated for the overflow case [47], it has not been investigated for the underflow case. It is well known that queue underflow will result in low utilization of the available bandwidth in a network. In order to prevent the underflow, the STQ needs to be sized accordingly. In the standard, the maximum fairness round trip time (*maxFRTT*) is defined as the round trip time for propagation of a fairness value around an entire ring and for the first affected traffic to return to the congested station.

Denote N as the total number of stations on a ringlet. Let *advertisingInterval* be the interval that each station advertises its own *addRate,* and *ringKM* be the circumference of the ringlet, then *maxFRTT* can be calculated as shown in Equation (3.15). Note that the constant 5μs is used as the propagation delay of a signal per km of the medium.

$$maxFRTT = N * advertisingInterval + 2 * (5\mu s * ringKm) \tag{3.15}$$

Note that *maxFRTT* does not account for the total delay for the sizing of the STQ to prevent underflow. Another major component results from the fairness algorithm of RPR. When a congested station is no longer congested, it will start advertising *FULL_RATE* to indicate the absence of congestion. The source station will then start incrementing its *allowedRate* up to a maximum rate defined as *LINK_RATE*. The *allowedRate* is incremented according to Equation (3.16).

$$\begin{aligned} allowedRate = allowedRate + \\ (LINK\_RATE - allowedRate) / rampUpCoef \end{aligned} \tag{3.16}$$

Define agingInterval as the interval a source station increments its own allowedRate and rampUpCoef as an arbitrary constant. Denote the additional delay before a station reaches its maximum rate of "LINK_RATE" as *rampUpDelay*. The *rampUpDelay* can then be calculated according to Equation (3.17).

$$rampUpDelay = \frac{agingInterval * LINK\_RATE}{rampUpCoef} \tag{3.17}$$

The impact of oscillations on link utilization can be resolved by correct sizing of the STQ for the underflow case. To prevent underflow of the STQ, the queue needs to be sized so that it cannot be emptied before the feedback control loop takes effect. Therefore, after the congested station declares that it is not congested anymore (which is defined as the queueSize being less than lowThreshold), it should have enough buffer build-up in order to transmit the sum of maxFRTT and rampUpDelay.

$$lowThreshold > (maxFRTT + rampUpDelay) * lineRate \tag{3.18}$$

Figure 3.26 shows the results of rerunning the hubscenario shown in Figure 3.21 by using the guideline according to Equation (3.18) for correct sizing of the STQ.

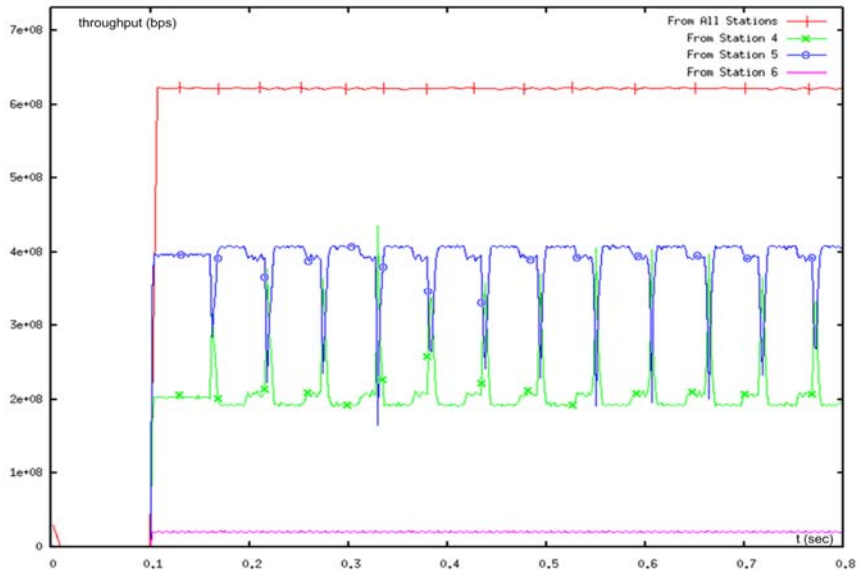

Figure 3.26 Throughput vs. time of the scenario with buffer threshold at Station *6* adjusted for underflow.

The oscillatory behavior of the total traffic received from all stations is not there anymore and the link utilization is at 100%. There are periodic interruptions to the traffic sourced from Stations *4* and *5*. The oscillations occur as a result of the already buffered traffic at the upstream stations while these buffers are being depleted during traffic adjustment periods.

As long as a system has enough buffering and the traffic can tolerate jitter, one can utilize the additional buffer for the fairness eligible packets and prevent oscillations at the destination to provide maximum utilization of the network. Alternatively, if the buffers are not available at the MAC client, one can always utilize the mechanism described in the previous section via weighted fairness parameters to completely eliminate oscillations.

3.8 Destination-Based Fair Dropping -- Active Queue Management for MAC Client Implementation of Resilient Packet Rings

Virtual destination queuing as discussed previously provides higher utilization of the ring. It incurs higher complexity in terms of the number of queues that needs to be supported as well as the scheduling algorithm that needs to be implemented. As an alternative it is feasible to implement an active queue management algorithm similar to Approximate Fair Dropping algorithm [48]. This section details proposed algorithm Destination Based Fair Dropping (DBFD) [49].

3.8.1 RPR Fairness Distribution

The standard defines two methods to distribute the fairness information around the ring. The first method is used to distribute the fair rate of the nearest congested station to the upstream stations. This fairness message is called single-choke fairness frame (SCFF) in the standard. The second message is used to propagate the fair rate of each station to all the other stations. This fairness message in RPR is called multi-choke fairness frame (MCFF). Each station on the ring puts its own congestion status (which gives how much its output link is used by the station itself) in such a message and sends it to all the other stations on the ring. A receiving station may collect these messages, and then builds a global image of the congestion situation on the ring, and schedule the traffic to add to the ring accordingly. Figure 3.27 shows the separation of RPR MAC and its client. The RPR MAC transfers MCFF and SCFF messages via the control path indication; while the MAC client transmits and receives the packets via data path request and indication messages, respectively. MAC Datapath Sublayer handles the transmission and reception of the frames to and from the dual ringlets.

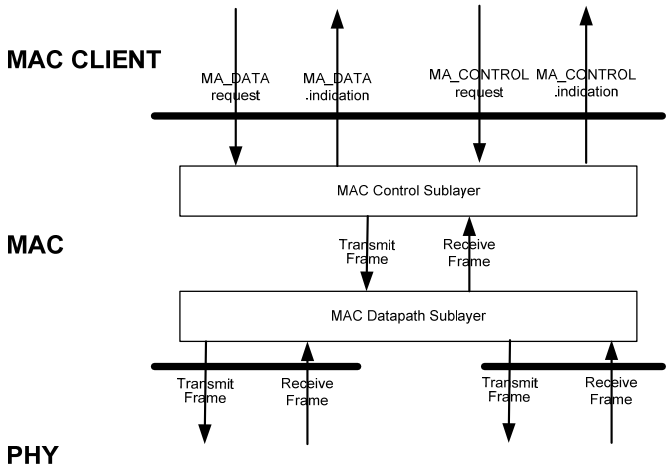

Figure 3.27 RPR MAC services model.

The implementation details of the RPR MAC client layer is not part of the standard; however, Appendix J of the standard [1] shows an example for a single queue implementation that utilizes SCFF and another one that utilizes MCFF with virtual output queues. By shaping traffic according to the MCFF messages at the MAC client, one can increase the bandwidth utilization by avoiding single congestion points. This section describes an RPR MAC client implementation that utilizes a modified Approximate Fair Dropping (AFD) algorithm [48]. The modified algorithm is referred to as Destination based Fair Dropping (DBFD).

3.8.2 Multi Destination Traffic Scenario

In this section, an example scenario as shown in Figure 3.28 is investigated. In this scenario, Stations *2*, *3*, *4* and *5* have traffic destined to Station *1*. Meanwhile Station *5* has also traffic destined to Stations *1*, *3* and *4*.

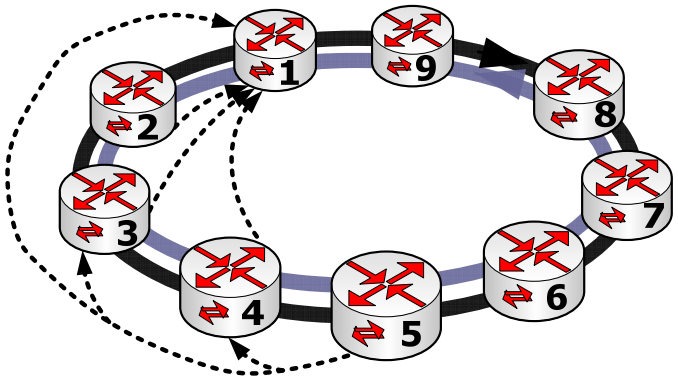

Figure 3.28 Multi destination scenario.

Stations *2*, *3*, and *4* start sending traffic to Station *1* at time 1sec. Station *5* also starts sending traffic to Stations *1* and *4* at time 1sec. Station *5* then starts sending traffic to Station *3* at time 2sec. Note that the traffic demand of each session at each station is OC12 rate per each session from one station to another.

Per RIAS fairness [35], the 620Mbps bandwidth on the link between Stations *1* and *2* should be equally shared resulting in 155Mbps per station. Station *5* can utilize more bandwidth without impacting this fairness. There is additional 465Mbps bandwidth on the link between Stations *5* and *4* and also a maximum available bandwidth of 310Mbps on the link between Stations *4* and *3*. In the ideal case, Stations *1*, *3* and *4* should receive 620Mbps, 232.5MBps and 232.5Mbps, respectively. Therefore, the total bandwidth utilization on the ring will be 1085Mbps. This scenario will be used to compare the performance of different MAC client implementations in the following sections.

3.8.3 RPR MAC Client with DBFD

Most of the active queue management techniques (e.g., RED [50]) utilize the queue size to make a drop decision on each packet arrival. DBFD is similar to the other active queue management schemes in that it also uses a FIFO queue size with probabilistic drop-on-arrival. DBFD not only relies on past measurements of the queue size but also recent observed rates of flows. By using this additional information, DBFD can provide fairness among different flows [48].

One can approximate a virtual output queuing scheme by using a single FIFO queue with active queue management as will be shown in Section 3.8.8. The implementation of DBFD in RPR requires a modification of the AFD algorithm since in RPR the fair rate also changes the drop probabilities of all frames destined to all stations after the station (excluding the station itself) which sent the fair rate. The MAC client will actively adjust the drop probability of each packet to each destination by using the fair rate. If a fair rate with congestion information is received from a station on the ring, all

the frames destined after that station will have increased probability of being discarded. While providing fairness among destinations, the MAC client will not have to implement 255 destination queues with the DBFD algorithm. This scheme will require per destination counters in MAC client (which is already required in a multi-queue implementation in the standard). Thus, the hardware implementation will be simplified or microcode based implementations may be deployed.

3.8.4 Algorithm of RPR MAC Client with DBFD

Consider a ring with the source Station *s* and the destination Station *d* as shown in Figure 3.29. On this ring, assume Station *i* has the minimum fair rate in between Station *s* and Station *d*. Also define Station *j* as any arbitrary downstream station beyond Station *s*.

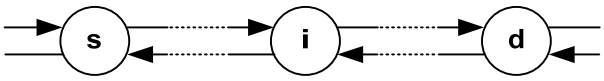

Figure 3.29 Station with minimum fair rate between Stations *s* and *d*

On this ringlet, denote $\mathbf{F}_s$ as the received fair rate vector at Station *s*, where $\mathbf{F}_s$ is the set of fair rates obtained from SCFF or MCFF sent by all the downstream stations in between Station *s* and Station *d* at time *t*. Then, f_i is the received fair rate at Station *s* from Station *i* where

$$f_i = \min\left(\mathbf{F}_s\right) \tag{3.19}$$

Define a flow vector as $\mathbf{R}=\{r_{sj}\}$, in which a flow from Station *s* to Station *j* is denoted by r_{sj}. By using the above definitions, the total traffic sourced by Station *s* that is destined beyond Station *i* is

$$k_s(i) = \sum_{\forall j:\ Station\ j > Station\ i} r_{sj} \tag{3.20}$$

Define Q_{ref}, α_1, α_2 as arbitrary constants, $dbfd'_s$ as the previous value of the DBFD rate, q_s as the length of the queue at Station s in bytes and q'_s as the previous value of the queue length, then the current DBFD rate at Station s can be computed as:

$$dbfd_s = \begin{cases} Q_{ref} & q(s) = 0 \\ dbfd'_s - \alpha_1(q_s - Q_{ref}) + \alpha_2(q'_s - Q_{ref}) & q(s) \neq 0 \end{cases} \quad (3.21)$$

Define β as an arbitrary constant and r_{sd} as the rate of flow from Station s to Station d, then the drop probability of a packet destined to Station d will be given as:

$$p_s(d) = \begin{cases} 1 & f_i < k_s(i) \\ 1 - f_i / k_s(i) & (f_i \geq k_s(i)) and (r_{sd} \geq \beta * dbfd_s) \\ 0 & (f_i \geq k_s(i)) and (r_{sd} < \beta * dbfd_s) \end{cases} \quad (3.22)$$

According to the above drop probability, if the allowed rate is less than the total traffic sourced by Station s destined to Station d, then the traffic destined to that station will be dropped. If the traffic destined to Station d still has more DBFD fair rate allowed, the traffic will not be dropped. This will provide differentiation between the traffic flows destined to Station d and the traffic flows destined beyond Station d. Finally, if the allowed rate is still higher than the total rate destined to Station d, then the traffic will be dropped probabilistically.

3.8.5 RPR MAC Client Implementation with DBFD

Figure 3.30 provides the sample code that will be executed when a new fairness message is received at Station s. To carry out this calculation, the station will need to keep an array of 255 counters, which is also required in all multi-choke fairness algorithms.

```
//Fairness message from Station i received
If(rcvd_usage[i] != NO_CONGESTION) {
      F[i]=rcvd_usage[i]);
}
```

Figure 3.30 Code snippet to execute when a fairness message is received.

Figure 3.31 provides the sample code that shows calculations required at each RPR parameter calculation interval called decay interval.

```
// Allowed usage updated at each decay interval
// MAX_STATIONS on RPR ring is 255
for (j=0; j<=MAX_STATIONS; j++) {
  allowd = ((LINK_CAPACITY -
                    F[j])/ LP_ALLOW_FACTOR);
  F[j] += allowd;
}

sum0 = (LP_COEFF-1.0) * lp_usage + tot_usage;
if (sum0 >= 1.0) {
   lp_usage = sum0 / LP_COEFF;
} else {
   lp_usage = 0.0;
}
if (tot_usage >= 0.5) {
  tot_usage  = tot_usage -
                    (tot_usage / AGE_COEF);
} else {
  tot_usage = 0.0;
}

// Usage aged at each decay interval
for (int j=0; j<=MAX_STATIONS; j++) {
  r[j] = r[j] -
          (r[j] / AGE_COEF);
}

qlen_old = qlen;
qlen = get_queue_length();

// DBFD rate calculated at each decay interval
if (qlen == 0) {
  dbfd_fair = Q_REF;
} else {
  dbfd_fair = dbfd_fair - a1 * (qlen - QREF)
                  + a2 * (qlen_old - QREF);
}
if (dbfd_fair <0) {
  dbfd_fair = 0;
}
```

Figure 3.31 Code snippet to execute at each decay interval.

This process does the low-pass filtering of internal counters so that the system does not respond to sudden changes immediately in order to provide stabilization in the fairness algorithm. It has been shown in [38] that the LP_COEFF and AGE_COEFF are

the two parameters that directly affect the stability of the RPR fairness algorithm with respect to the size of the ring. The main addition is the calculation of the "dbfd_fair" rate at each decay interval on top of the standard algorithm.

```
//Packet destined to Station d received. Decide
//if it is okay to queue the packet at Station s.
// MAX_STATIONS on RPR ring is 255
i= 0; ki = 0; congestion_station = 0;
r_max = LINK_CAPACITY;
while (i<d) {
  if (F[i] < (double)LINK_CAPACITY) {
   // There is a possible congestion
   // calculate the max allowed rate
   if (F[i] < r_max) {
     r_max = F[i];
     congestion_station = i;
    }
   }
   i++;
  }
}
i = congestion_station;
fi =r_max;
for (j=i+1;j<=MAX_STATIONS;j++){
  ki  += r[j];
}
if (ki > fi) {
  pd = 1.0;
} else if ( ((r[i]) < beta *dbfd_fair) ) {
  pd = 0.0;
} else {
  pd = (1 - dbfd_fair/(r[i]));
}
rdm = rand()/RAND_MAX;
if (pd <= rdm) {
  r[d] += pktByte + HEADER_OVERHEAD;
 // Okay to queue the packet
} else {
 // do not accept the packet to the queue
}
```

Figure 3.32 Code snippet to execute at each packet arrival.

Figure 3.32 shows the code snippet that gets executed at packet arrival destined to Station *d*. Among all three code pieces shown in Figures 3.30, 3.31 and 3.32 the one in 3.32 requires the highest computational complexity and needs to be efficient. The "while loop" in Figure 3.32 can be simplified by performing the calculations when a fairness message is received. In addition, the total usage (ki) can be kept separately instead of calculating it each time. All the calculations have the complexity of $O(n)$; however, if there is not enough processing power, one can employ the sampling algorithm proposed

in [48]. This approach allows rate estimations at certain intervals so that one does not burden the system with calculating the rates at each packet arrival.

The current single queue implementation of RPR may result in overly underutilized rings in some scenarios and oscillations can also be observed in those scenarios [35]. With the proposed mechanism, the ring utilization can easily approach the theoretical limit of the product of number of links and the bandwidth with a relatively simple implementation. The advantage of this algorithm for RPR is that it improves the performance of an RPR ring and it is backward compatible with the standard as compared to the previously proposed solutions. In addition, the idea does not require 255 independent queues to be implemented in the scheduling hierarchy. Adding an additional level to the scheduling hierarchy is not possible without requiring new hardware.

3.8.6 Simulation Results with Single Queue

The scenario is simulated using the RPR model implemented in the OPNET simulator. An OC12 ring which is composed of nine stations is created with 20km of distance between each adjacent station. Each station is configured as a dual-queue station with the aggressive fairness mode enabled. The size of the secondary transit queue (STQ) at each station is 512KB and the "LP_COEFF" [1] parameter of the RPR MAC is set to 4.

Figure 3.33 shows the total traffic sourced by Stations *2*, *3*, *4* and *5* to the outer ringlet. As expected, the available bandwidth is being shared equally by Stations *2*, *3*, and *4*, while Station 5 is able to get more bandwidth out of the ring by utilizing the unused bandwidth on the links.

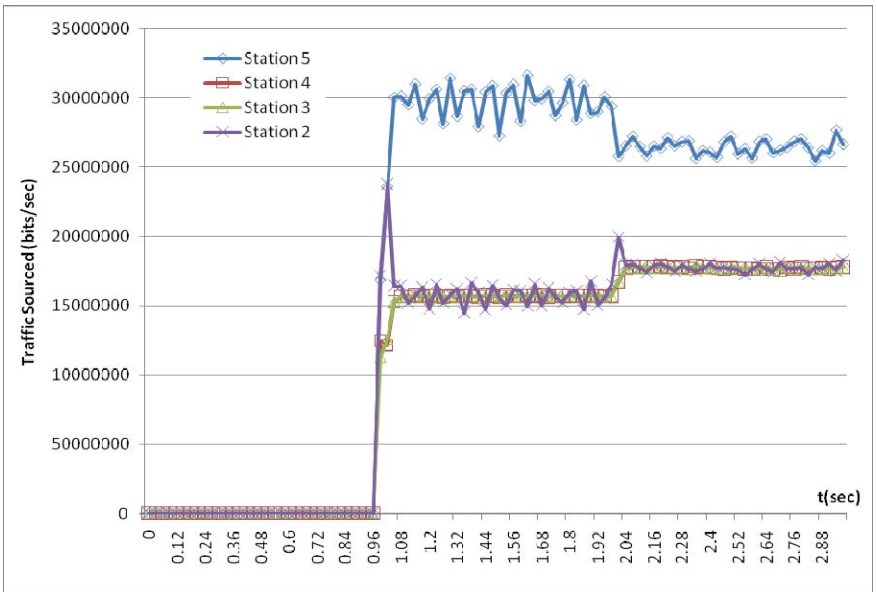

Figure 3.33 Actual traffic sourced at Stations 2, 3, 4 and 5.

As shown in Figure 3.34, even though Station *1* receives the full 620Mbps of traffic, Station *5* is not able to utilize the full unused bandwidth.

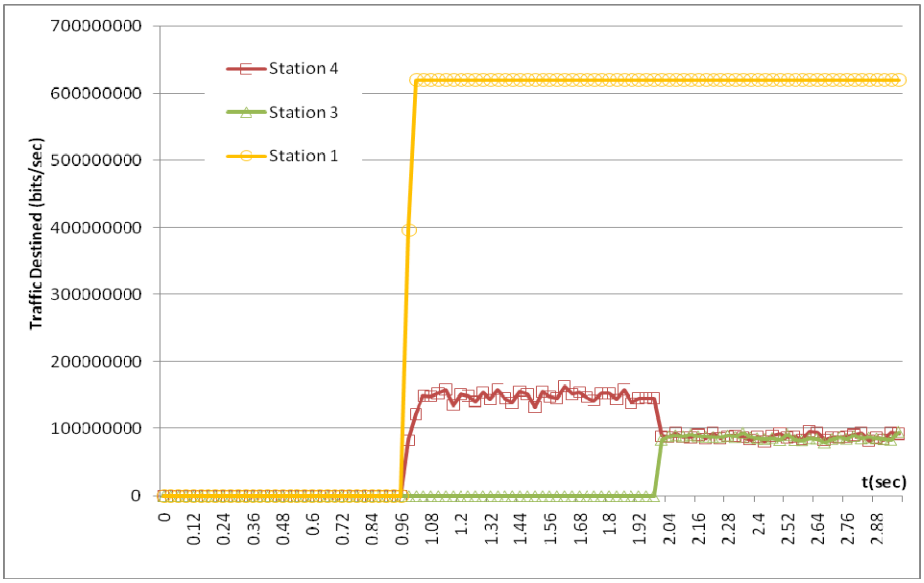

Figure 3.34 Traffic received at Stations 2, 3 and 4.

In addition, once Station *5* starts sending to Station *3* at time 2sec, the fairness is lost and is not able to get its fair share of the bandwidth, and Stations *2*, *3* and *4* start sourcing more traffic to Station *1* than Station *5*. The total ring utilization is 820Mbps

instead of the expected 1085Mbps. Note that oscillations are observed around the steady state.

3.8.7 Simulation Results with Multiple Queues

The RPR MAC client model with virtual output queues is implemented as explained in the standard. Figure 3.35 shows the actual traffic sourced at Stations *2*, *3*, *4*, and *5*. In this case, the oscillations are minimized, and the steady response is observed at time 2 sec, when the Station *5* starts sending traffic to Station *3*.

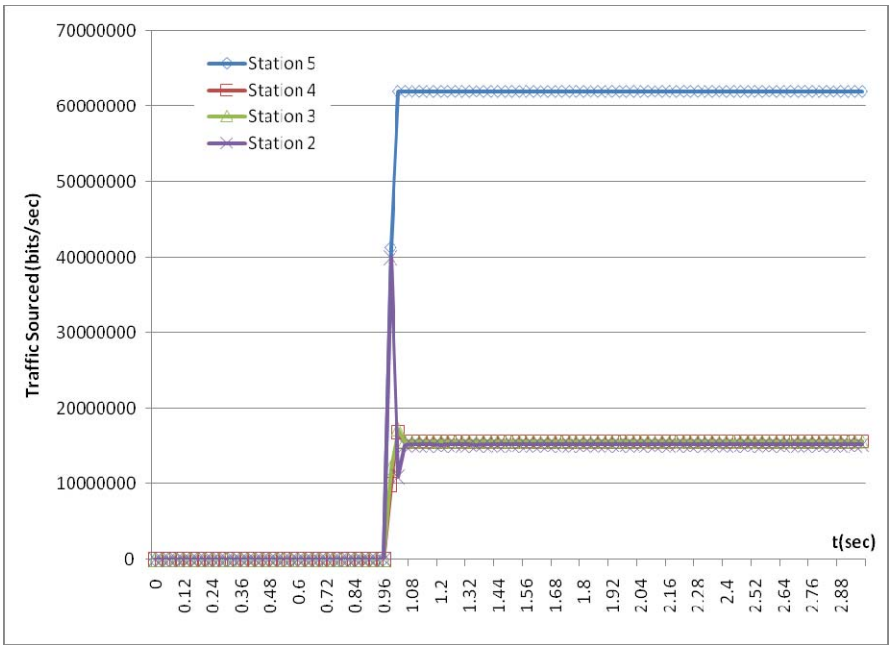

Figure 3.35 Actual traffic sourced at Stations 2, 3, 4 and 5.

In addition, once Station *5* starts sending traffic to Station *3* at time 2sec, the fairness is lost and is not able to get its fair share of the bandwidth, and Stations *2*, *3* and *4* start sourcing more traffic to Station *1* than Station *5*. The total ring utilization is 820Mbps instead of the expected 1085Mbps. Also note that oscillations are observed around the steady state.

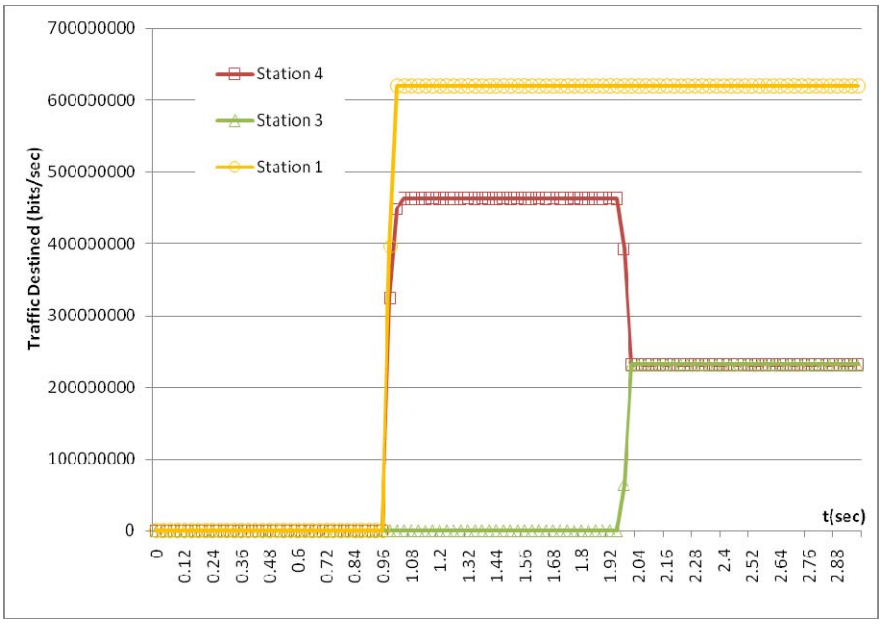

Figure 3.36 Traffic received at Stations 2, 3 and 4.

3.8.8 Simulation Results with DBFD

The RPR MAC client model with DBFD is also implemented as explained in Section 3.8.5. Figures 3.37 and 3.38 show the actual traffic sourced and received at various stations.

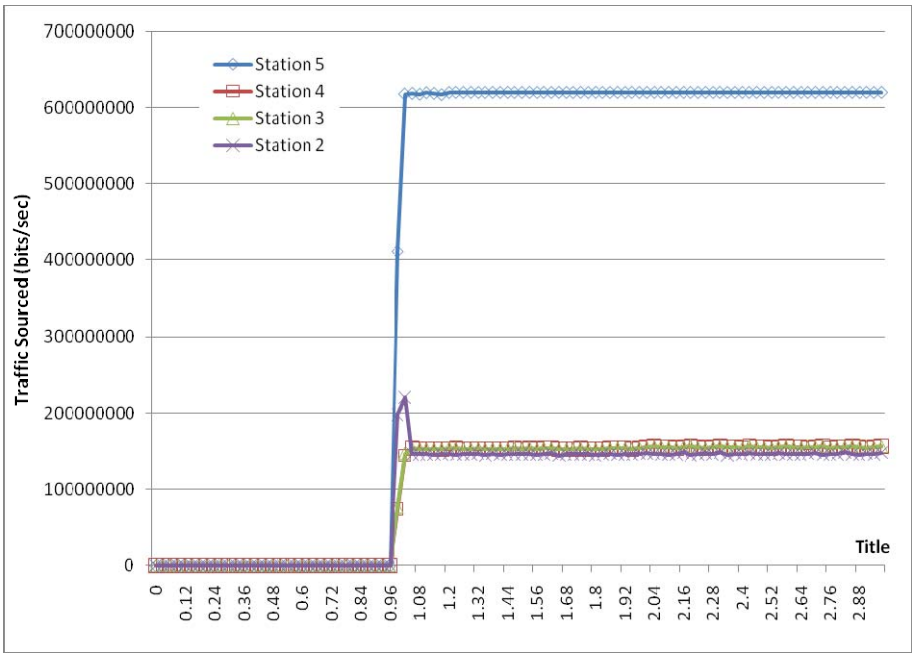

Figure 3.37 Actual traffic sourced at Stations 2, 3, 4 and 5.

Again, the oscillations are minimized and steady response is observed at time 2 sec, when Station *5* starts sending traffic to Station *3*. The performance is similar to the behavior observed in the multiple queue implementation of the MAC client.

Destination based fair dropping algorithm provides an efficient mechanism to handle multi-choke fairness in an RPR network. The same mechanism can also be extended to be used in any network where destination stations provide congestion status information. As shown above, while preserving fairness among stations, this approach has improved the utilization of the underlying network as compared to the single queue implementation of the standard. In addition, this approach does not require any modifications to the standardized IEEE 802.17

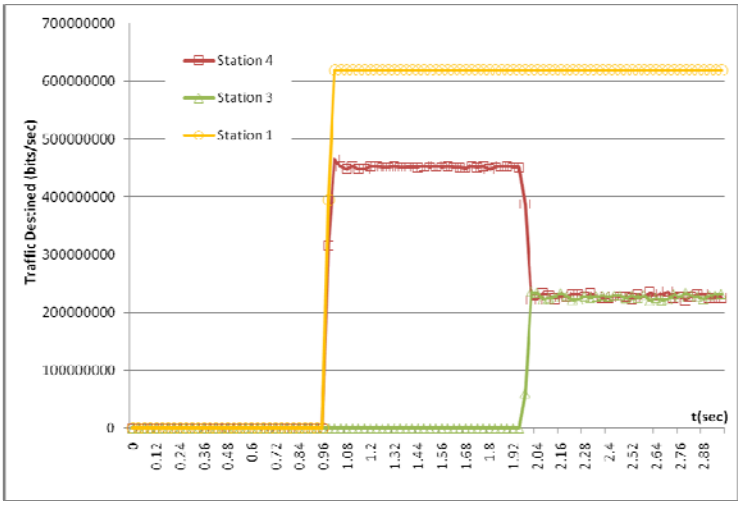

Figure 3.38 Traffic received at Stations 2, 3 and 4.

3.9 Weighted Destination-Based Fair Dropping

As shown in wRIAS with destination differentiation definition, it is possible to distinguish the destination flows at each station. The previous DBFD algorithm does not support this. In this section the weighted destination based fair dropping algorithm (wDBFD) [51] is proposed.

The implementation of wDBFD takes into account of the received fair rate from a congested station. This fair rate changes the packet drop probabilities of all flows destined to stations downstream to that congested station (excluding the congested station itself) once the aggregated rate of flows exceeds the received fair rate. Therefore, the MAC client needs to actively adjust the drop probability of each packet to each destination by considering the received fair rates from congested stations. While providing fairness among destinations, the MAC client will not need to implement 255 destination queues with the wDBFD algorithm. Instead, this scheme utilizes per destination counters in the MAC client (most of which are already necessary for a multi-queue implementation of the standard). Thus, the hardware implementation is simplified and allows microcode based implementations on presently deployed hardware. As compared to the DBFD algorithm, wDBFD requires an additional ingress counters per destination.

For the wDBFD algorithm, consider a ring with the source Station *s* and the destination Station *d* as shown in Figure 3.39. On this ring, assume Station *i* has the minimum fair rate in between Station *s* and Station *d* and denote Station *j* as any arbitrary downstream station beyond Station *s*.

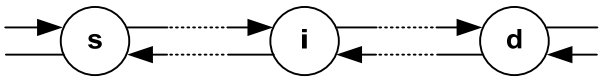

Figure 3.39 Station with minimum fair rate between Stations *s* and *d*.

On this ringlet, $\mathbf{U}_s$ is denoted as the received fair rate vector at Station *s*, where $\mathbf{U}_s$ is the set of fair rates obtained from SCFF or MCFF sent by all the downstream stations in between Station *s* and Station *d* at time *t*. If u_i is the minimum of all fair rates received,

then Station *i* is the most congested station in between the source Station *s* and the destination Station *d*.

$$u_i = \min\left(\mathbf{U}_s\right) \tag{3.23}$$

Define a flow vector, $\mathbf{F}=\{f_{sd}\}$, in which the arrival rate of a flow to Station *s* destined to Station *d* is denoted by f_{sd}. Define another flow vector, $\mathbf{R}=\{r_{sj}\}$, in which a flow sourced by Station *s* destined to Station *j* is denoted by r_{sj}. By using the above definitions, the total traffic sourced by Station *s* that is destined beyond Station *i* is given by Equation (3.24).

$$k_s(i) = \sum_{\forall j:\ Station\ j > Station\ i} r_{sj} \tag{3.24}$$

Define Q_{target}, α_1, α_2 as arbitrary constants, $dbfd'_s$ as the previous value of the wDBFD rate, q_s as the length of the queue at Station *s* in bytes, and q'_s as the previous value of the queue length, then the current wDBFD rate at Station *s* is given by Equation (3.25).

$$dbfd_s = \begin{cases} Q_{target} & q(s) = 0 \\ dbfd'_s - \alpha_1(q_s - Q_{target}) + \alpha_2(q'_s - Q_{target}) & q(s) \neq 0 \end{cases} \tag{3.25}$$

Define β as an arbitrary constant to normalize the wDBFD rate with respect to the rate measurement and f_{sd} as the arrival rate of a flow at Station *s* destined to Station *d*, then the drop probability of a packet destined to Station *d* is given by Equation (3.26).

$$p_s(d) = \begin{cases} 1 & w_s \cdot u_i < k_s(i) \\ 1 - \dfrac{\rho_{sd} \cdot \beta \cdot dbfd_s}{f_{sd}} & (w_s \cdot u_i \geq k_s(i)) and (f_{sd} \geq \rho_{sd} \cdot \beta \cdot dbfd_s) \\ 0 & (w_s \cdot u_i \geq k_s(i)) and (f_{sd} < \rho_{sd} \cdot \beta \cdot dbfd_s) \end{cases} \tag{3.26}$$

As defined in Equation (3.26), if the received fair rate (u_i) is less than the total traffic sourced by Station *s* destined beyond Station *i*, then the traffic destined beyond that Station *i* will be dropped. If the Station *s* is still allowed to send more traffic beyond Station *i*, then based on the wDBFD rate and the arrival rate (f_{sd}) the traffic may be randomly dropped. The drop probability goes up if the arrival rate of a flow is much higher than the wDBFD rate. If a flow continues to send at the higher rate, it will be penalized more by increasing the drop probability. This allows the algorithm to be more stable with respect to different packet arrival rates. If the arrival rate of traffic is higher than the departure rate from the queue, the algorithm will operate at buffer occupancy of Q_{target}. The algorithm will establish a certain packet mix in the buffer so that the ratio of packets destined to each station will correspond to the ratio of allowed fair rates to each destination. In addition, the drop probabilities are also adjusted by weight ρ_{sd} to allow different destinations to receive different proportions of the available bandwidth.

Similar to the DBFD algorithm, the advantage of this algorithm for RPR is that it improves the performance of an RPR ring, and it is backward compatible with the standard which is not true for the previously proposed solutions. In addition, it does not require 255 independent queues to be implemented in the scheduling hierarchy. In general adding an additional level to the scheduling hierarchy is not possible without requiring new hardware.

The implementation of the wDBFD algorithm requires additional calculations on top of the current RPR fairness. Figure 3.40 provides the sample code that will be executed when a new fairness message is received by Station *s*. To carry out this

calculation, the station will need to keep an array of 255 counters which is also required in all multi-choke fairness algorithms.

```
//Fairness message from Station i received
If(rcvd_usage[i] != NO_CONGESTION) {
      u[i]=rcvd_usage[i]);
}
```

Figure 3.40 Code snippet to execute when a fairness message is received.

Figure 3.41 provides the sample code that shows calculations required at each RPR parameter calculation interval called decay interval.

```
// Allowed usage updated at each decay interval
// MAX_STATIONS on RPR ring is 255
for (j=0; j<MAX_STATIONS; j++) {
  allowd = ((LINK_CAPACITY -
                  u[j])/ LP_ALLOW_FACTOR);
  u[j] += allowd;
}

sum0 = (LP_COEFF-1.0) * lp_usage + tot_usage;
if (sum0 >= 1.0) {
   lp_usage = sum0 / LP_COEFF;
} else {
   lp_usage = 0.0;
}
if (tot_usage >= 0.5) {
  tot_usage  = tot_usage -
                  (tot_usage / AGE_COEF);
} else {
  tot_usage = 0.0;
}

// Usage aged at each decay interval
for (int j=0; j<MAX_STATIONS; j++) {
  r[j] = r[j] –
          (r[j] / AGE_COEF);
}

// Arrival rate aged at each decay interval
for (int j=0; j<MAX_STATIONS; j++) {
  f[j] = f[j] –
          (f[j] / AGE_COEF);
}


qlen_old = qlen;
qlen = get_queue_length();

// DBFD rate calculated at each decay interval
if (qlen == 0) {
  dbfd_fair = Q_TARGET;
} else {
  dbfd_fair = dbfd_fair - a1 * (qlen - QREF)
                  + a2 * (qlen_old - QREF);
}
if (dbfd_fair <0) {
  dbfd_fair = 0;
}
```

Figure 3.41 Code snippet to execute at each decay interval.

This process does the low-pass filtering of internal counters so that the system does not respond to sudden changes immediately in order to provide stabilization in the fairness algorithm. It has been shown in [38] that the LP_COEFF and AGE_COEFF directly affect the stability of the RPR fairness algorithm with respect to the size of the ring. The main addition is the calculation of the "dbfd_fair" rate at each decay interval on top of the standard algorithm.

Figure 3.42 shows the code snippet that gets executed at a packet arrival destined to Station *d*. This incurs the highest computational complexity and needs to be efficient as it gets executed at each packet arrival.

```
//Packet destined to Station d received. Decide
//if it is okay to queue the packet at Station s.
// MAX_STATIONS on RPR ring is 255
i= 0; ki = 0; congestion_station = 0;
r_max = LINK_CAPACITY;

while (i<d) {
  if (u[i] < (double)LINK_CAPACITY) {
   // There is a possible congestion
   // calculate the max allowed rate
   if (u[i] < r_max) {
     r_max = u[i];
     congestion_station = i;
    }
   }
   i++;
  }
}
i = congestion_station;
ui = r_max;
for (j=i+1;j<=MAX_STATIONS;j++){
  ki  += r[j];
}
if (ki > w[s] * ui) {
  pd = 1.0;
} else if ((f[i] < beta * ro[s,d] *dbfd_fair) ) {
  pd = 0.0;
} else {
  pd = (1 – beta * ro[s,d] * dbfd_fair/(f[i]));
}
rdm = rand()/RAND_MAX;
f[d] += pktByte + HEADER_OVERHEAD;
if (pd <= rdm) {
  r[d] += pktByte + HEADER_OVERHEAD;
 // Okay to queue the packet
} else {
 // do not accept the packet to the queue
}
```

Figure 3.42 Code snippet to execute at each packet arrival.

The "while loop" in Figure 3.42 can be simplified by performing calculations when a fairness message is received up front. In addition, the total usage (ki) can be tracked separately instead of calculating it each time a packet is received. When the DBFD rate is calculated at each decay interval, the values adjusted by "ro(s,d)" can also be calculated up front and stored in a table to be used at each packet arrival. All the calculations have the complexity of $O(n)$. If there is not enough processing power, one can employ the sampling algorithm proposed in [48]. This approach allows rate estimations at certain intervals so as not to burden the system with calculating the rates at each packet arrival.

3.9.1 Performance Evaluation of Weighted Destination Based Dropping

The example scenario shown in Figure 3.28 will be used to compare the performance of different MAC client implementations. In this scenario, Stations *2*, *3*, *4* and *5* have traffic destined to Station *1*. Station *5* has also traffic destined to Stations *3* and *4*. Stations *2*, *3*, and *4* start sending traffic to Station *1* at time 1sec. Station *5* starts sending traffic to Stations *1* and *4* at time 1sec. Station *5* then starts sending traffic to Station *3* at time 2sec. In this scenario, the traffic demand of all but one session at each station is OC12 rate per each session from one station to another. While the scenario resembles to the one given in Section 3.8.2, in this case, Station *5* receives two times more traffic to destination Station 3 than the other stations to demonstrate the stability of the wDBFD algorithm.

Per RIAS fairness [35], the 620Mbps bandwidth on the link between Stations *1* and *2* should be equally shared resulting in 155Mbps per station. Station *5* can utilize

more bandwidth without impacting this fairness. There is additional 465Mbps bandwidth on link between Stations *5* and *4* and also a maximum available bandwidth of 310Mbps on link between Stations *4* and *3*. In the ideal case, Stations *1*, *3* and *4* should receive 620Mbps, 232.5MBps, and 232.5Mbps, respectively. Therefore, the total bandwidth utilization on the ring will be 1085Mbps based on be "weighted" ingress aggregated fair with destination differentiation definition. This behavior expected to be the same as that in Section 3.8.2 will be used to compare the performance of different MAC client implementations.

The scenario is simulated using the single queue RPR model implemented in the OPNET simulator. An OC12 ring which is composed of nine stations is created with 20km of distance between every two adjacent stations. Each station is configured as a dual-queue station with the aggressive fairness mode enabled. The size of the secondary transit queue (STQ) at each station is 512KB and the "LP_COEFF" [1] parameter of the RPR MAC is set to 4.

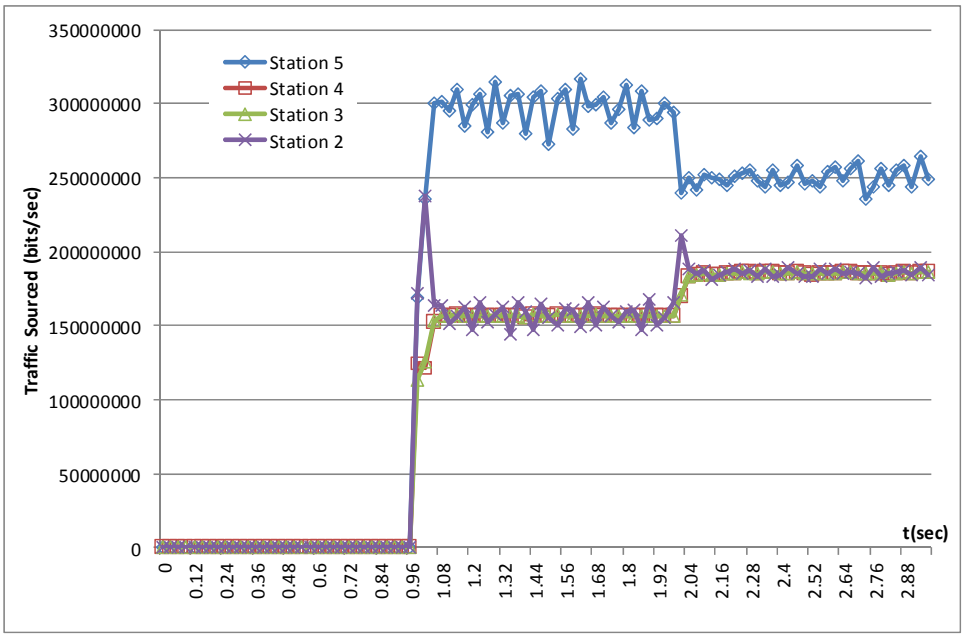

Figure 3.43 Actual traffic sourced at Stations *2, 3, 4* and *5* with single MAC client queue.

Figure 3.43 shows the total traffic sourced by Stations *2*, *3*, *4* and *5* to the outer ringlet. As expected, the available bandwidth is being shared equally by Stations *2*, *3*, and *4*, while Station *5* is able to get more bandwidth out of the ring by utilizing the unused bandwidth on the links. However, at time 2 second, once Station *5* starts sending traffic to Station *3,* the fairness message generated by Station *4* limits the total traffic that can be sourced by Station *5*. This is similar to the graph shown in Figure 3.33 as expected.

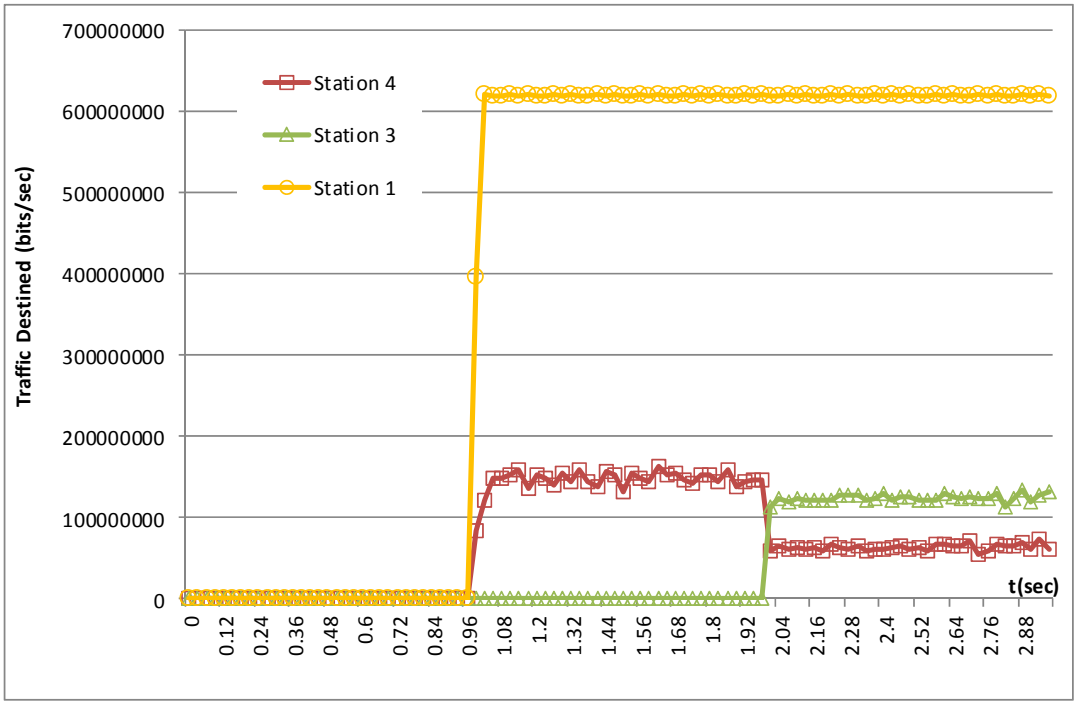

Figure 3.44 Traffic received at Stations 1, 3 and 4 with single MAC client queue.

Station *1* receives the full 620Mbps of traffic, while as shown in Figure 3.43 Station *5* is not able to utilize the unused bandwidth fully. Interestingly this time, once Station *5* starts sending packets to Station *3* at time 2sec, the fairness is lost and Station *5* is not able to get its fair share of the bandwidth, and Stations *2*, *3* and *4* start sourcing more traffic to Station *1* than Station *5*. The total ring utilization is 810Mbps instead of the expected 1085Mbps. One other observation is that after time 2 sec, Station *3* receives

more traffic than Station *4* even though the traffic is sourced by the same Station *5*. The main reason for this behavior is the imbalanced traffic demand used in this scenario and the simple single queue implementation not being able to maintain fairness at the source station per destination. Therefore, destination fairness is not achieved.

The same scenario is simulated with the RPR MAC client model using virtual output queues (VoQ) as explained in the standard. Figure 3.45 shows the actual traffic sourced at stations *2*, *3*, *4*, and *5*. In this case, the oscillations are minimized, and a steady response is observed at time 2 sec, when Station *5* starts sending traffic to Station *3*.

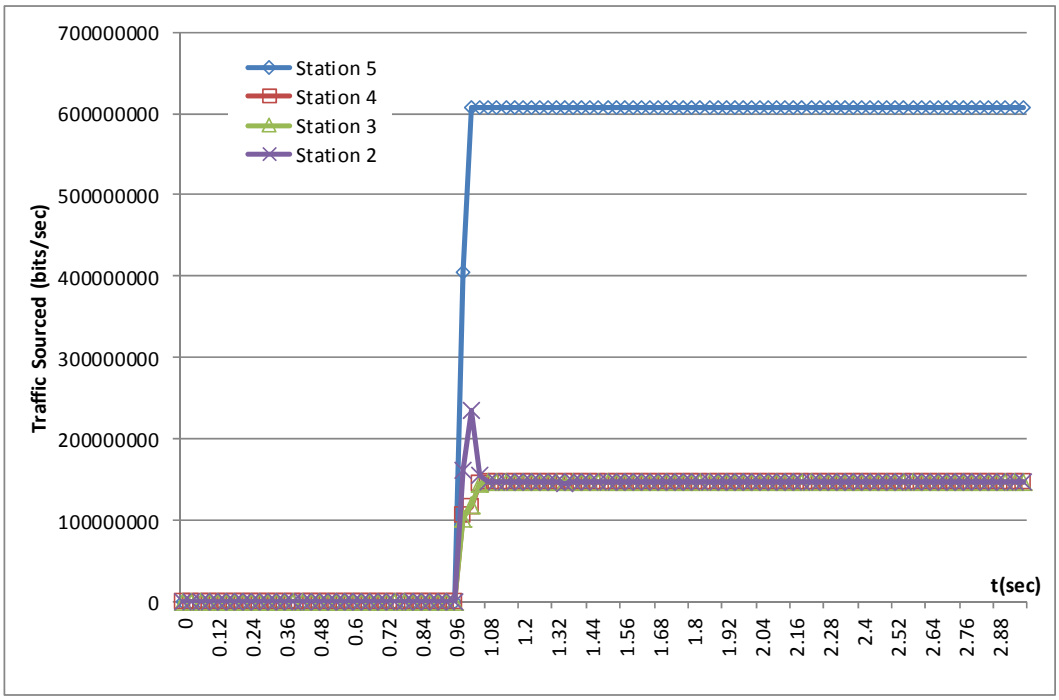

Figure 3.45 Actual traffic sourced at Stations 2, 3, 4 and 5 with VoQ.

Figure 3.46 shows the traffic received at Stations *2*, *3,* and *4*, respectively. The observed bandwidth matches the expected values, and provides a maximum bandwidth utilization of 1085Mbps. In addition, the destination fairness is achieved with respect to traffic received at Station *3* and Station *4*, since virtual output queuing is able to maintain destination separation with respect to different packet arrival rates for each destination.

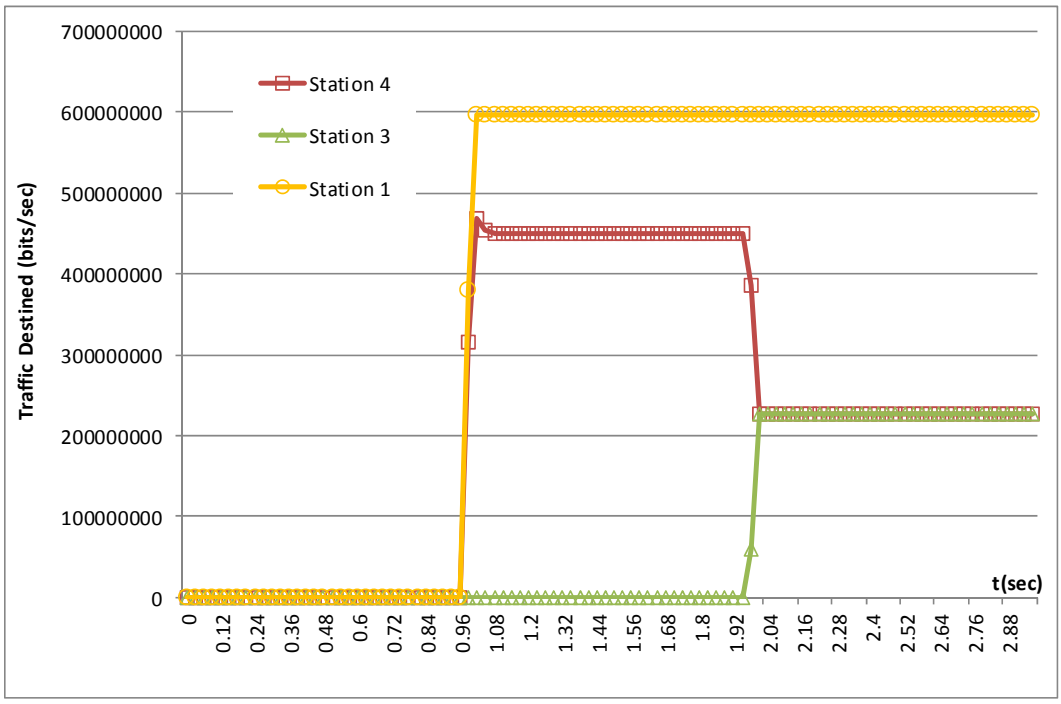

Figure 3.46 Traffic received at Stations 1, 3 and 4 with VoQ.

Next, the same scenario is simulated using the RPR MAC client with DBFD as explained in Section 3.8. Figures 3.47 and 3.48 show the actual traffic sourced and received from and at respective stations. The steady response is observed after time 2 sec when Station *5* starts sending traffic to Station *3* and the total ring utilization reaches up to the expected 1085Mbps. Similar to the single queue RPR implementation, the destination fairness is not achieved for the traffic sourced by Station *5* to the destination Stations *3* and *4*. Specifically, Station *3* receives almost two times more traffic than Station *4*. Since the packet arrival rate destined to Station *3* after time 2 sec is two times more than the packet arrival rate destined to Station *4* and the arrival rate is not regulated per destination, this undesirable behavior is expected for the DBFD algorithm as well.

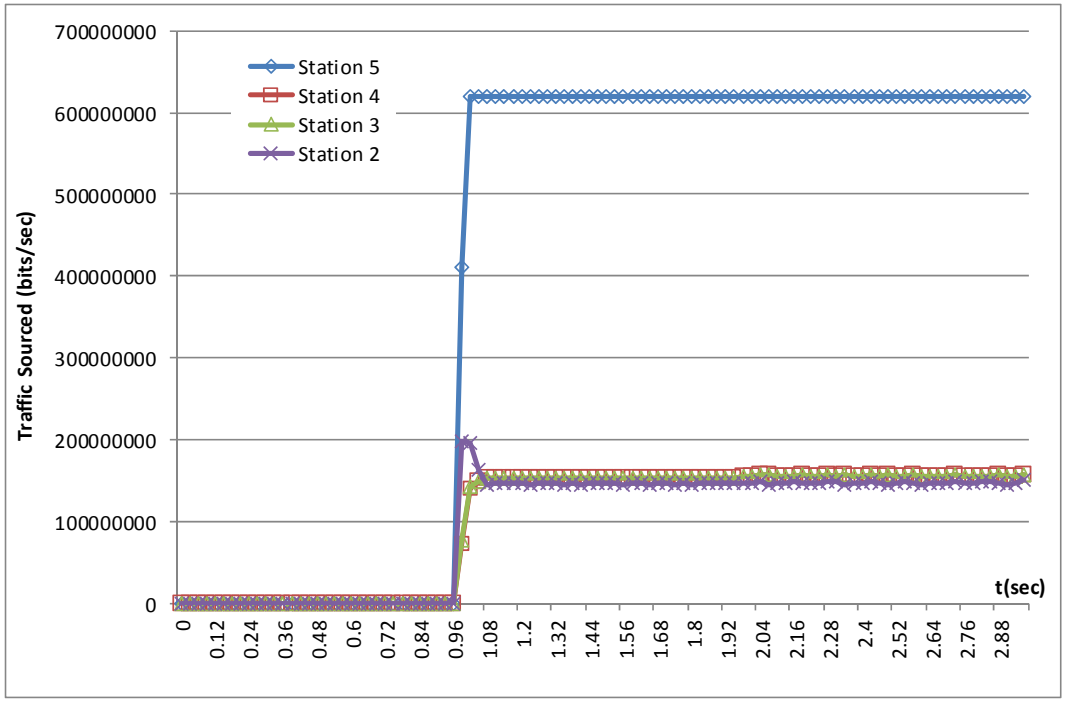

Figure 3.47 Actual traffic sourced at Stations 2, 3, 4 and 5 with DBFD.

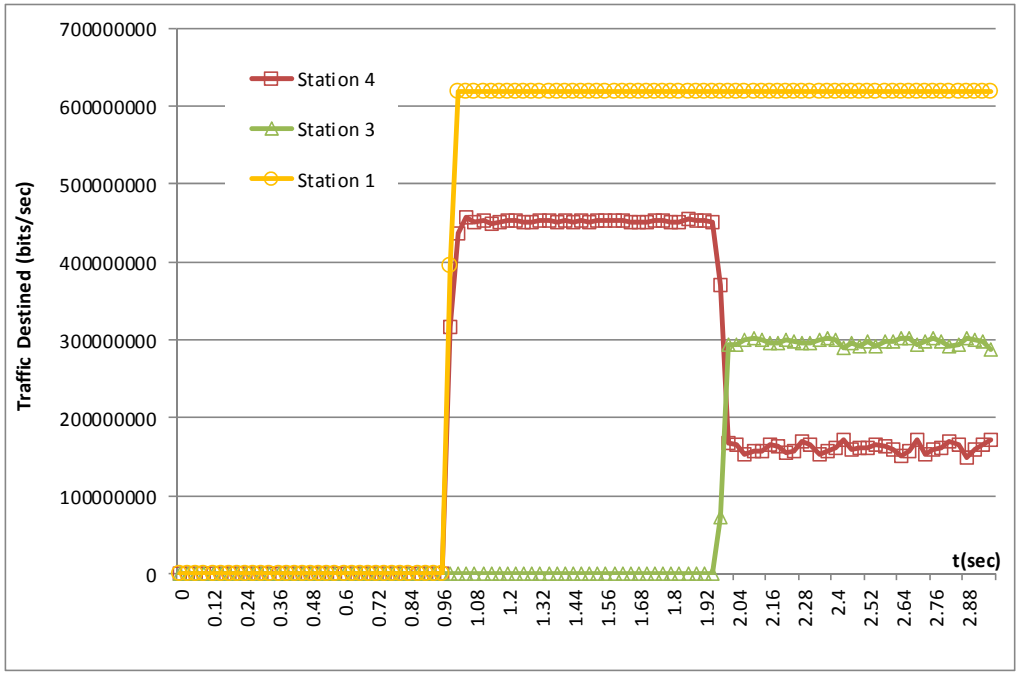

Figure 3.48 Traffic received at Stations 1, 3 and 4 with DBFD.

Next, the same scenario is simulated with the wDBFD algorithm as given in Section 3.9. The results are shown in Figures 3.49 and 3.50.

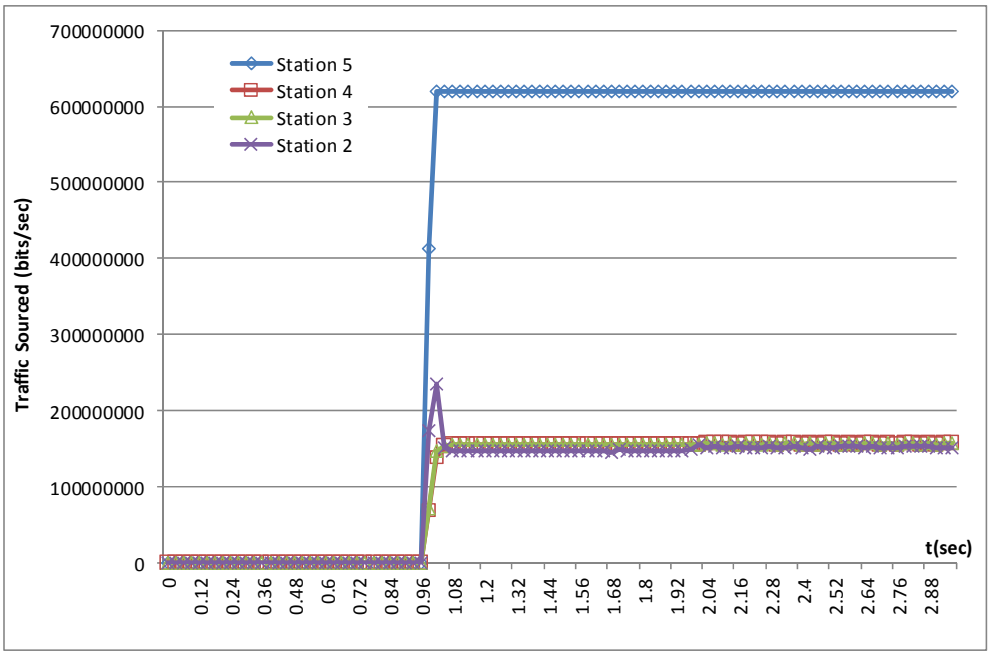

Figure 3.49 Actual traffic sourced at Stations 2, 3, 4 and 5 with wDBFD.

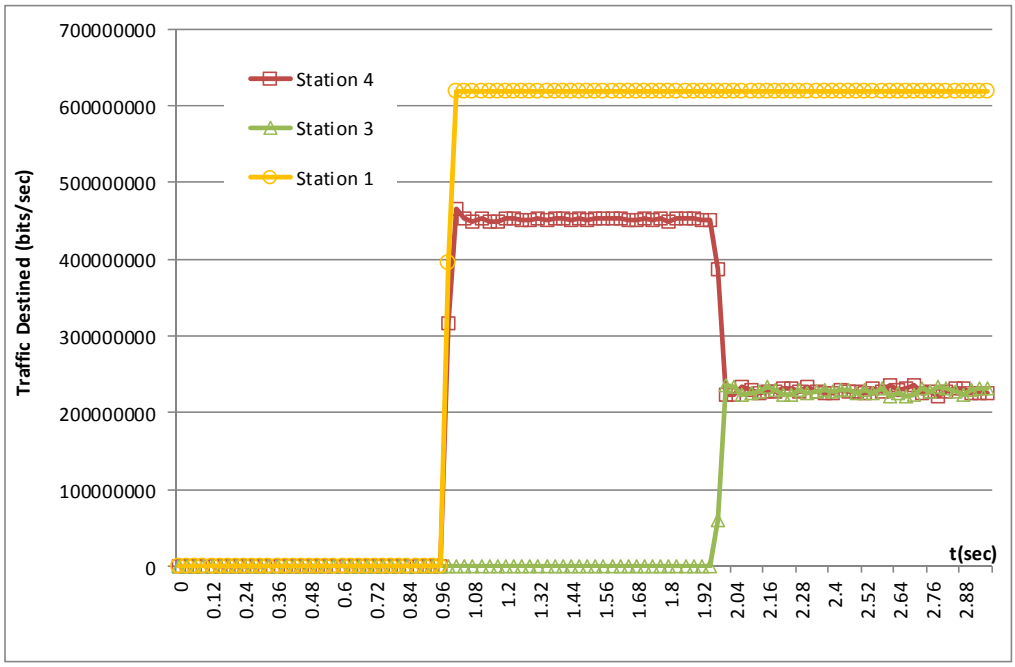

Figure 3.50 Traffic received at Stations 1, 3 and 4 with wDBFD.

The results shown in Figures 3.49 and 3.50 are very similar to the ideal case with virtual output queues. Destination fairness at Station *5* is maintained regardless of its packet arrival rate difference from destination Stations *3* and *4*. This is made possible by

increasing the probability of packet drop per flow according to the ratio of the arrival rate of a flow to the "DBFD" rate. This provides stability and fair sharing of the queues even when the arrival rates and/or packet sizes are different among flows.

Another aspect to consider is the convergence speed. The interval required for the ring traffic to converge to the fair rate is also impacted by the desired queue occupancy (Q_{target}). Based on the RPR fairness messages and the destination weights, the MAC client queue will have the right mix of packets to match the fair rates. As the desired queue occupancy increases, the convergence to fair rates will take longer. If the desired queue occupancy (Q_{target}) is set too low, unnecessary packet drops can be observed. This scenario has been tested with different packet arrival patterns per destination. Regardless of the packet arrival rates and packet sizes, the destination stations receive similar amount of traffic as shown in Figures 3.49 and 3.50.

3.9.2 Providing Destination Differentiation Through wDBFD

The proportion of traffic destined to stations can be adjusted by the ρ_{sd} parameter as described in Section 3.9. This adjustment relies on adjusting the drop probabilities of flows per destination with respect to each other as well as buffer occupancy. The ρ_{sd} parameters will dictate the ratio of packets in the MAC client queue destined to different stations. In this section, the scenario is modified such that the destination Station *4* is given a weight of 2 (ρ_{54}=2) while the flows destined to Stations *1* and *3* are each assigned to weight of 1.

Figures 3.51 and 3.52 show the simulation results with the destination weight adjustment. In this case, the Station *5* can send two times more traffic to Station *4* than to the other destinations. Specifically, as shown in Figure 3.52, the destination Station *5* is

able to transmit approximately 310Mbps of traffic to Station *4*, while it transmits 155Mbps of traffic to the Stations *1* and *3*. This shows that the algorithm can efficiently provide destination differentiation as required even when the amount of traffic destined to Station *3* is much higher than the amount of traffic destined to Station *4*.

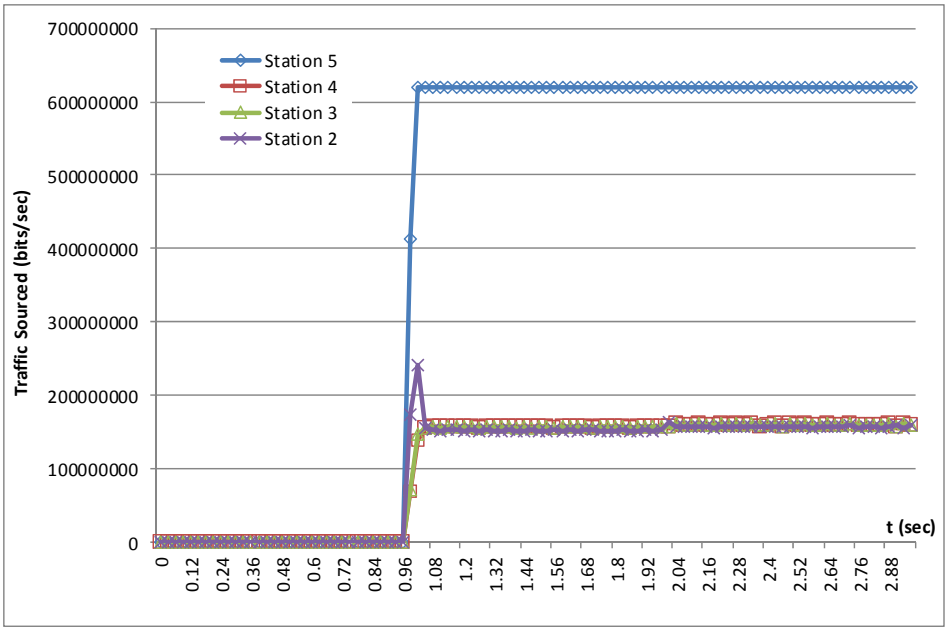

Figure 3.51 Actual traffic sourced at Stations 2, 3, 4 and 5 with wDBFD.

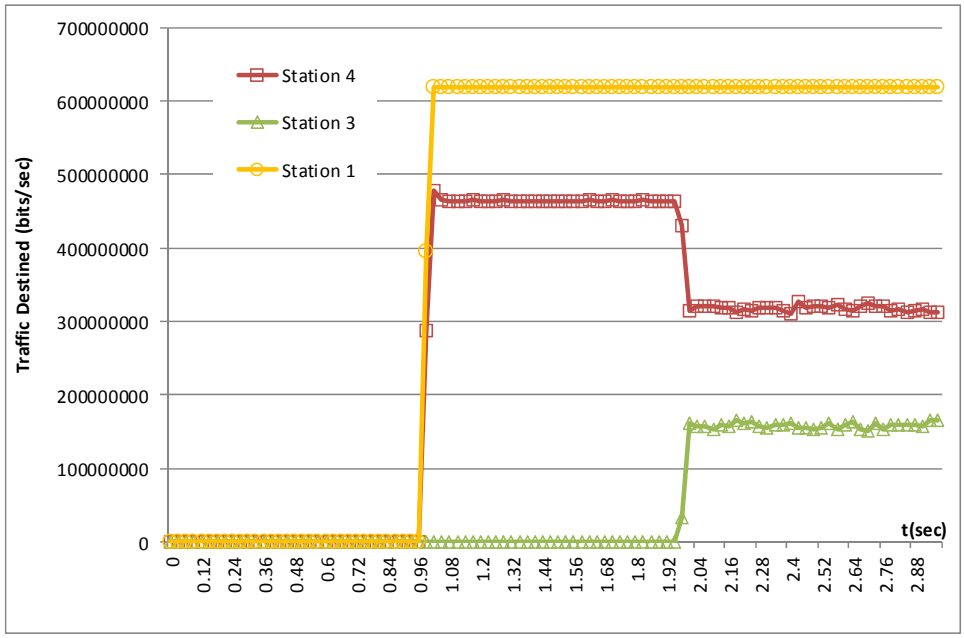

Figure 3.52 Traffic received at Stations 1, 3 and 4 with wDBFD.

3.10 Resilient Packet Rings with Heterogeneous Links

The RPR standard calls for links with the same capacity to be used to establish a ring. There will be, however, cases where it may not make financial sense for a service provider to set up a ring with uniform link capacities especially when the provider wants to deploy its ring over an existing network. In addition, in some cases, parts of the MAN may not be as densely populated as other sections, and hence the service provider may choose to deploy lower speed links in those parts. This aspect of RPR has not been studied before. In this section, supporting non-uniform link capacity in RPR networks [52] is presented.

From the fairness algorithm point of view, a non-uniform link capacity is not desired as the standard fairness algorithm relies on single congestion point identification on the ring. When there are lower capacity links, the ring utilization and spatial reuse will decrease considerably. The standard allows sending packets to the stations that are located before the congested station. Unfortunately, it is affected by the head of line blocking, and hence its performance depends on the arrival rate of the packets destined to different stations.

RPR provides a means to overcome the head of line blocking issue by passing detailed ring congestion information to the MAC clients, hence allowing more advanced clients. A generic way to utilize this mechanism fully is to implement 255 separate queues (one queue per destination) within the MAC client as shown in [49]. Clearly, this is an expensive way in terms of the MAC layer hardware to support such feature. To resolve this issue, one can utilize the weighted destination based fair dropping (wDBFD) algorithm as shown in the previous section.

Note that the standard defines the fair rate information to be represented in 16-bits. This is a normalized representation of the bandwidth based on the link speeds. Therefore, when a ring is established with non-uniform links, the fairness algorithm should use the link with the maximum capacity as a normalization factor on all stations. This is the basis of the algorithm to allow all stations to interoperate when heterogeneous links are present.

The scenario shown in Figure 3.53 will be used to compare the performance of different MAC client implementations in the presence of non-uniform links. The links that are marked as OC3 are the slower capacity links while the rest of the links are OC12 as shown in Figure 3.53. In this scenario, Station *5* has traffic destined to Stations *1, 3* and *4*. Meanwhile, Station *3* has traffic destined to Stations *2*. Note that the traffic from Station *3* traverses OC3 link.

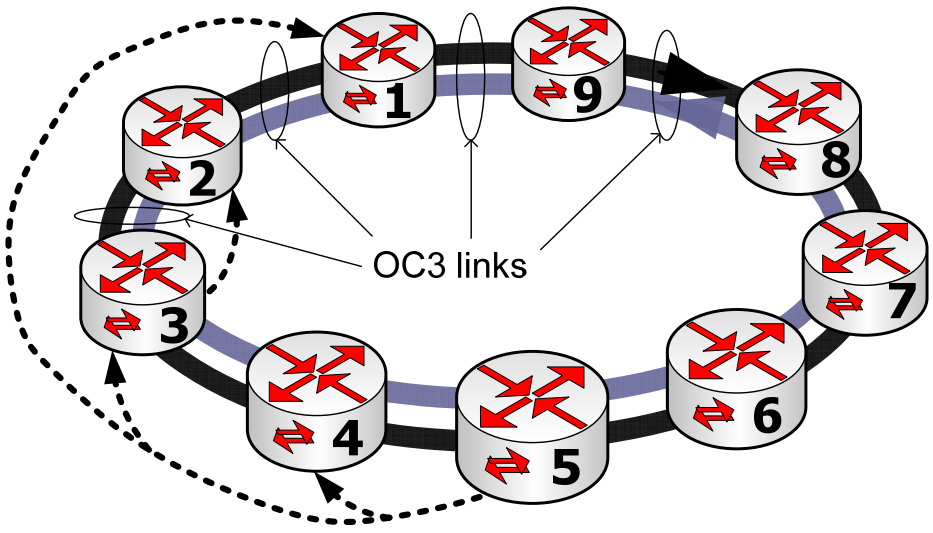

Figure 3.53 Multi destination scenario with non-uniform links.

Stations *3* starts sending traffic to Station *2* at time 1sec. Station *5* also starts sending traffic to Stations *3* and *4* at time 1sec. Station *5* then starts sending traffic to

Station *1* at time 2sec. Note that the traffic demand of each session at each station is OC12 rate per each session from one station to another.

Per RIAS fairness [35], the 155Mbps bandwidth on the link between Stations *2* and *3* should be equally shared resulting in 77Mbps per station. Station *5* can utilize more bandwidth without impacting this fairness. There is additional 572Mbps bandwidth on the link between Stations *5* and *4*. In the ideal case, Stations *1, 2*, *3* and *4* should receive 77Mbps, 77Mbps, 271MBps, and 271Mbps, respectively. Therefore, the total theoretical bandwidth utilization on the ring is 696Mbps for this scenario after Station *5* starts transmitting to Station *1* at time 2sec. This scenario will be used to compare the performance of different MAC client implementations in the following sections in order to evaluate the operation of RPR on different capacity links.

3.10.1 Simulation Results with Single Queue

The scenario is simulated by using the RPR model implemented in the OPNET simulator. The ring is composed of nine stations and each link between the stations covers 20km of distance. Each station is configured as a dual-queue station with the aggressive fairness mode enabled. The size of the secondary transit queue (STQ) at each station is 512KB and the "LP_COEFF" [1] parameter of the RPR MAC is set to 4.

Figure 3.54 shows the total traffic sourced by Stations *3* and *5* to the outer ringlet. As expected, the available bandwidth is being used up by Stations *3* and *5* as there is no major congestion point from time 1sec to 2sec. Once Station *5* starts transmitting packets beyond Station *2,* there is a big drop in the network utilization. Note that Station *1* is in the OC3 domain of the ring.

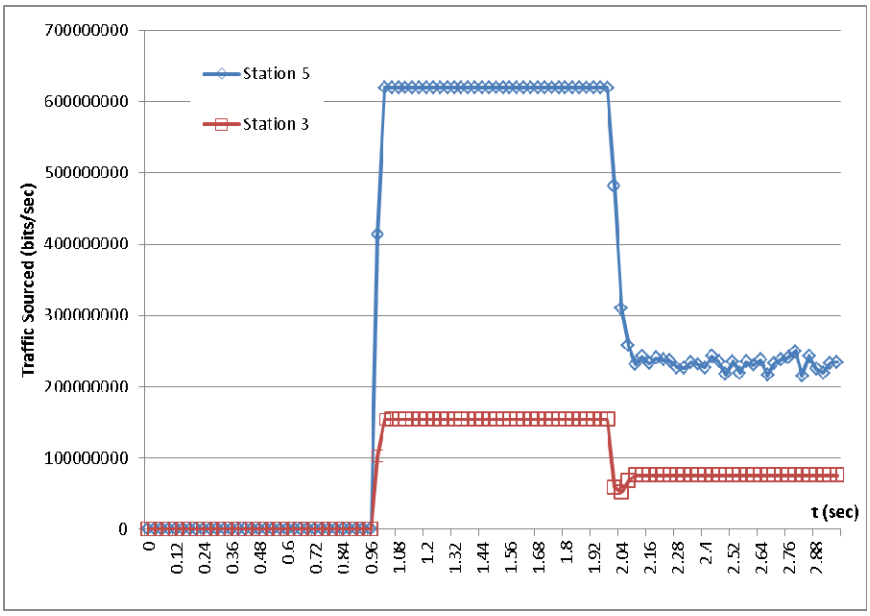

Figure 3.54 Actual traffic sourced at Stations 3 and 5.

As shown in Figure 3.55, all stations are receiving the maximum bandwidth before time 2sec. After Station *5* starts sending, they all stabilize and receive around 77Mbps. Stations *3* and *5* bottleneck at the same link once Station *5* starts sending to Station *1*. Based on RPR fairness, the OC3 link will be shared equally, which is approximately 77Mbps of throughput. Even though Stations *3* and *4* are out of the congestion domain, they are still limited to 77Mbps of throughput as well.

The packets destined to Stations *1*, *3* and *4* arrive at the same rate to Station *5* based on the scenario definition. This means that the ratio of packets waiting to be transmitted in the queue for each destination will be close to each other. In other words for each packet transmitted to Station *1*, there will approximately be one packet destined to Station *3* and another packet to Station *4*; when RPR MAC cannot accept any more packets destined to Station *1*, there will be head of line blocking in the single queue mechanism. Therefore, the total ring utilization for this scenario is only 308Mbps instead of the expected 696Mbps.

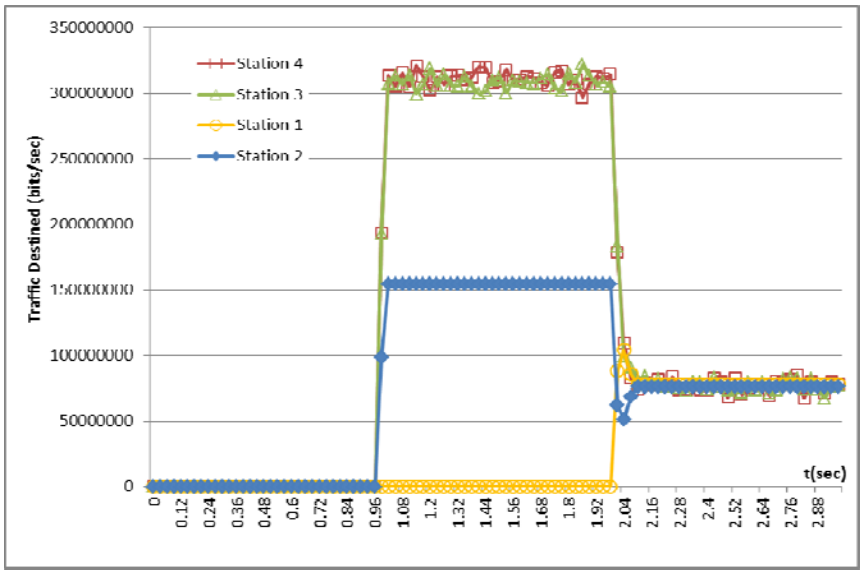

Figure 3.55 Traffic received at Stations 1, 2, 3 and 4.

Note that some oscillations are observed around the steady state. The reason of this behavior is associated with the fairness distribution in RPR. The SCFF identifies single most congested station on the ring, and that message will be propagated to the upstream stations as long as there is upstream traffic at a station. In this case, the most congested station, Station *3,* announces its fair usage information as 77Mbps from time to time and the upstream stations limit their traffic to that rate. Based on the arrival pattern of the packets, the traffic sourced by Station *5* will oscillate as the exact order in random arrivals can be different from the one-by-one arrival pattern of packets.

3.10.2 Simulation Results with VoQ

The same traffic scenario is run with the RPR MAC client model that implements virtual output queues as explained in the standard. Figure 3.56 shows the actual traffic sourced at Stations *3* and *5*. In this case, the oscillations are minimized, and the steady response is observed at time 2sec, when Station *5* starts sending traffic to Station *1*.

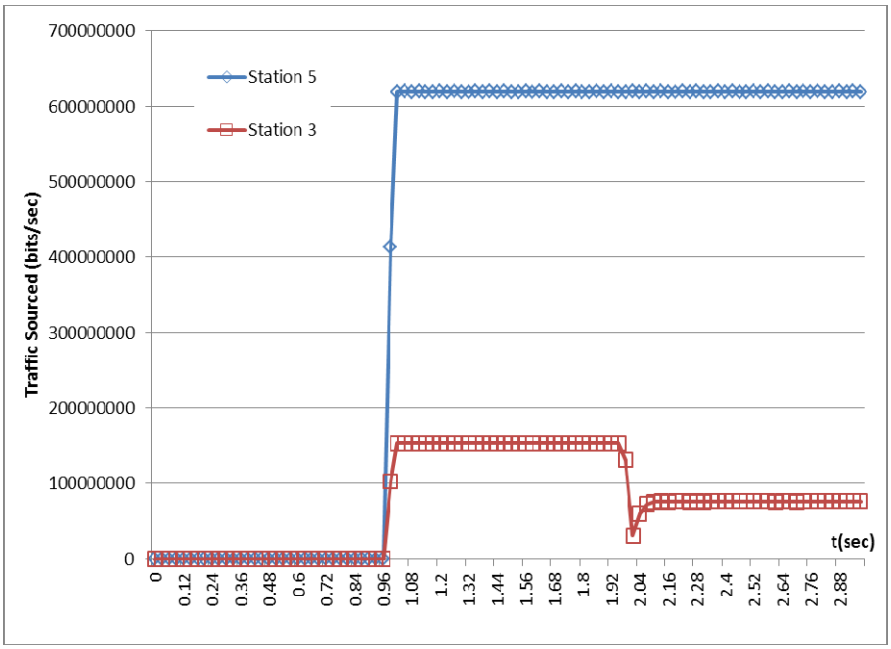

Figure 3.56 Actual traffic sourced at Stations 3 and 5 with VOQ.

Figure 3.57 shows the traffic received at Stations *1*, *2*, *3* and *4*, respectively. The total bandwidth reaches 696 Mbps. This total bandwidth matches with maximum achievable bandwidth under fairness constraints as calculated previously in Section 3.10.

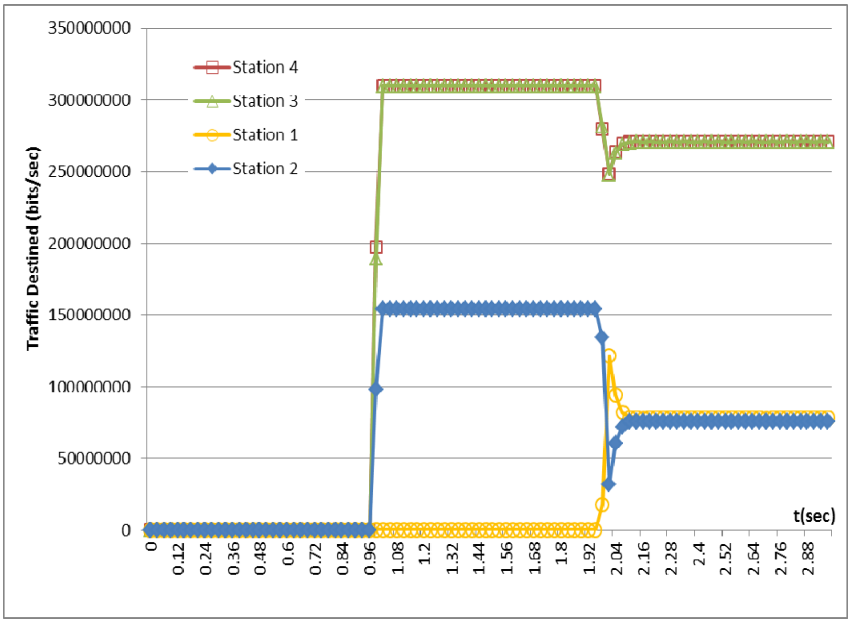

Figure 3.57 Traffic received at Stations 1, 2, 3 and 4 with VOQ.

Figures 3.58 and 3.59 show the actual traffic sourced and received at stations respectively with a wDBFD MAC client. Again, the oscillations are minimized and steady response is observed at time 2.2 sec, after Station *5* starts sending traffic to Station *1*. The performance is similar to the behavior observed in the multiple queue implementation of the MAC client.

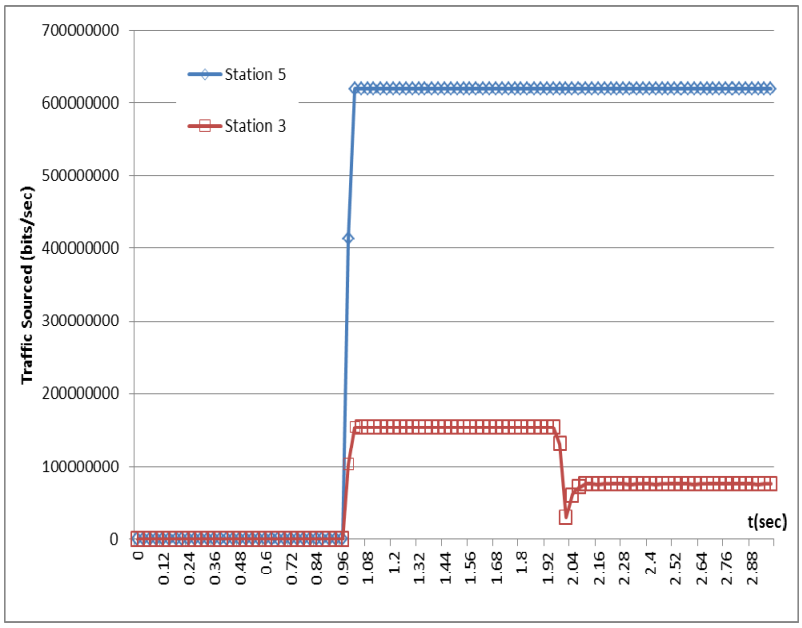

Figure 3.58 Actual traffic sourced at Stations 3 and 5 with wDBFD.

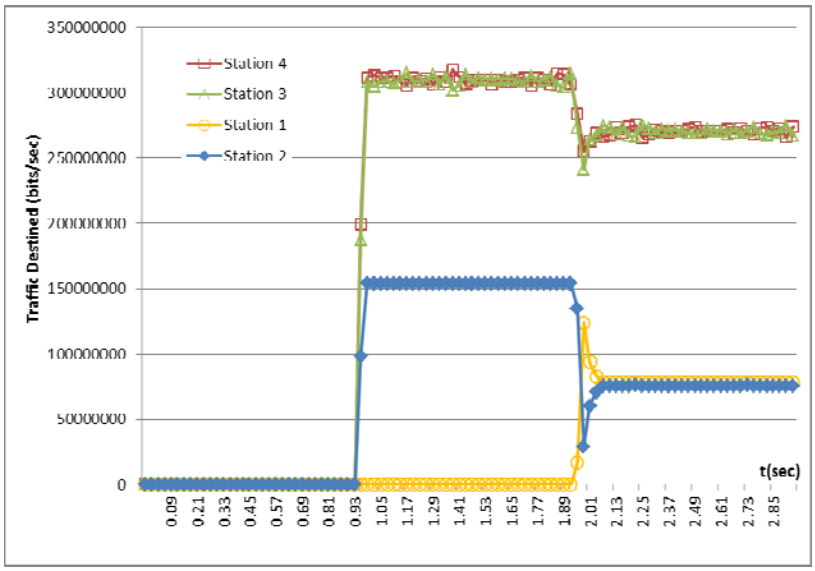

Figure 3.59 Traffic received at Stations 1, 3 and 4 with wDBFD.

Supporting non-uniform capacity links in an RPR network has not been studied previously. It is shown that if the fairness messages are normalized based on the maximum link capacity on the ring, the stations can communicate the fairness information among each other. As demonstrated in the above analysis and results, the standard MAC client implementation results in low network utilization as the upstream stations will not be able to utilize the high capacity links between each other when there is a congested station connected with low capacity links. In order to overcome this problem, more advanced MAC clients are necessary. By using either virtual destination queuing or wDBFD mechanism, one can reach the maximum utilization that can be achieved even with the presence of non-uniform capacity links. As illustrated earlier, the wDBFD mechanism can also be applied to any network where destination stations provide congestion status information. As shown previously in Section 3.9.2, wDBFD has improved the total network utilization as compared to the single queue implementation of the standard, and it can also support different capacity links in an RPR network with higher ring utilization. In addition, the proposed algorithms do not require any modifications to the standardized IEEE 802.17 RPR fairness mechanism.

CHAPTER 4

CONCLUSIONS

As the network capacity increases faster than the processing capacity simple algorithms are needed to control the flow of the packets. With simpler switching algorithms one can design less expensive networks with better efficiency. In order to achieve the most utilization out of the available capacity the header sizes and the stacking of protocols need to be optimized. The IEEE 802.17 Resilient Packet Ring standard aims at resolving these issues by utilizing the underlying SONET based infrastructure. As reported in many other articles [35,37,38], the fairness algorithm has failed under some pathological scenarios as also shown in this dissertation by simulations. In addition an additional case that failed to perform correctly with weighted fairness has also been identified as shown in Section 3.2.

A number of algorithms have been proposed to improve the bandwidth utilization for these pathological scenarios [35,37,39,40,49]. Since the IEEE 802.17 fairness algorithm is standardized, it is not practical to modify the underlying algorithm. Hence, this dissertation has carefully identified the root causes of various failures and proposed new and innovative ways to achieve better utilization of the network in utilizing the standardized algorithm.

This dissertation has presented a comprehensive overview of multiple algorithms (SRP, RPR, GFP) and described SONET that formed the basis of these. The RPR architecture has been discussed in detail. This dissertation has also covered how RPR has adopted features from earlier ring based protocols, and combined them into a novel and

coherent architecture. Various issues such as the class based priority scheme, station design, fairness, and resilience have been discussed. Performance evaluations using the latest version of the draft standard demonstrate how the protocol behaves using different options.

RPR is a MAC-layer technology that may span into MANs and WANs. RPR can easily bridge to Ethernet, including access networks like EFM. Thus, RPR allows layer 2 switching far into the backbone network, if such large link layer networks turn out to be practical. RPR may also do switching in the backbone network, by letting an RPR ring implement virtual point-to-point links between the routers connected to the stations on the ring. RPR may differentiate traffic; so when used to implement IP links, it is able to help the IP routers implement the QoS aware communication that is needed in a network that carries multimedia traffic.

This dissertation has also discussed the use of weighted fairness in an RPR network. The definition of ring ingress aggregated fairness has been extended by incorporating weights and destination differentiation into the formulation. Performance evaluations by using the latest version of the IEEE 802.17 RPR standard have demonstrated how the bandwidth is shared using different weights. In particular, a pitfall has been identified and improvements are suggested to circumvent that pitfall as substantiated by the simulation results. In addition, by adjusting various parameters already available in the fairness algorithm of IEEE 802.17, one can eradicate the oscillatory behavior under certain scenarios. Furthermore, by providing right queue size for the network, the utilization can be improved as the feedback loop works more efficiently.

The implementation of a mechanism to handle multi-choke fairness in an RPR network is also presented. The same mechanism can also be extended to be used in any network where destination stations provide congestion status information. As shown in this dissertation, while preserving fairness among stations, this approach has improved the utilization of the underlying network as compared to the single queue implementation of the standard. In addition, this approach does not require any modifications to the standardized IEEE 802.17 RPR fairness mechanism and allows a simpler and computationally less intensive implementation than the generic implementation provided in the standard.

Efficient active queue management mechanisms that utilize multi-choke fairness frames in an RPR network are proposed. The wDBFD algorithm presented in this dissertation provides weighted RIAS while providing weighted destination based fairness. As compared to the earlier active queue implementation, the wDBFD algorithm provides better isolation of flows with respect to different arrival rates. While preserving fairness among stations, this approach has improved the utilization of the underlying network as compared to the single queue implementation of the standard. In addition, this mechanism does not require any modifications to the standardized IEEE 802.17 RPR fairness mechanism either. Similarly, it allows a simpler and computationally less intensive implementation than the generic multi queue implementation discussed in the standard.

Finally, the support of non-uniform capacity links in an RPR network has been investigated. It is shown that if the fairness messages are normalized based on the maximum link capacity on the ring, the stations can communicate the fairness

information among each other. However, this results in low network utilization as the upstream stations will not be able to utilize the high capacity links between each other when there is a congested station connected with low capacity links. By either utilizing the wDBFD algorithm or the virtual destination queueing, one can reach the maximum utilization that can be achieved even with the presence of non-uniform capacity links.

In summary, contributions of this dissertation are threefold. The first set is of contributions comprise of the addition of weighted fairness, virtual destination queuing, worst case jitter analysis of secondary transit queue and the mechanism to limit the forward rate for uncommitted traffic class in RPR during the development of the RPR standard. The second contribution is to provide better understanding of the control mechanism in the presence of long path delays as a result of congested buffers, which can be resolved by correctly sizing the transit queues in the network. The third is to design MAC client mechanisms in addition to the mechanisms that are already included in the standard to resolve the deficiencies of the fairness algorithm in specific scenarios. All of these points are achieved by building on top of the standard itself and utilizing the mechanisms currently present in the standard without having to change the standard, thus setting these contributions apart from the other contributions that require substantial modifications to the standard.

REFERENCES

1. *Resilient Packet Ring (RPR) Access Method and Physical Layer Specifications*, IEEE Standard 802.17, 2004.

2. *SONET Transport Systems: Common Generic Criteria*, Telcordia Standard GR-253-CORE, 2009.

3. Tektronix, Inc., "SONET Telecommunications Standard Primer", Tektronix Inc., Beaverton, OR, Tektronix Document 2RW-11407-2, Sept. 2009.

4. *SONET - Sub-STS-1 Interface Rates and Formats Specification*, ANSI Standard T1.105, 1996.

5. S. Gorshe, "Automatic Protection Switching Technology White Paper", PMC Sierra Inc., Burnaby, BC Canada, Application Note PMC-2050248, Feb. 2005.

6. W. Simpson, "The Point-to-Point Protocol (PPP)", IETF Network Working Group, RFC 1661, Daydreamer, July 1994.

7. J. Reynolds and J. Postel, "Assigned Numbers", IETF Network Working Group, RFC 1340, USC/Information Sciences Institute, July 1992.

8. G. McGregor, "The PPP Internet Protocol Control Protocol (IPCP)", IETF Network Working Group, RFC 1172, Merit, May 1992.

9. W. Simpson, "PPP in HDLC Framing", IETF Network Working Group, RFC 1662, Daydreamer, July 1994.

10. A. Malis and W. Simpson, "PPP over SONET/SDH", IETF Network Working Group, RFC 2615, June 1999.

11. PMC Sierra, Inc., "Mapping Point To Point Protocol Over The Entire SONET/SDH SPE or Over Sub rate Tributary SPE's", PMC Sierra Inc., Burnaby, BC Canada, Application Note 960725, June 1996.

12. P. Langner, "SDL Data Link Specification", Lucent Technologies, Murray Hill, New Jersey, Sept. 1998.

13. J. Carlson, P. Langner, E. Hernandez-Valencia, J. Manchester, "PPP over Simple Data Link (SDL) using SONET/SDH with ATM-like framing", IETF Network Working Group, RFC 2823, May 2000.

14. J. Carlson, E. Hernandez-Valencia, N. Jones, P. Langner, J. Manchester, "PPP over Simple Data Link (SDL) using raw light wave channels with ATM-like framing", IETF PPP Working Group, Draft ietf-pppext-sdl-pol-00, June 1999.

15. J. Anderson, J.S. Manchester, R. Anderson, and M. Veeraraghavan, "Protocols and Architectures for IP Optical Networking," *Bell Labs Technical Journal*, pp. 105-124, Jan.-Mar. 1999.

16. D. Tsiang and G. Suwala, "The Cisco SRP MAC Layer Protocol", IETF Network Working Group, RFC 2892, Aug. 2000.

17. Fredrik Davik, Mete Yilmaz, Stein Gjessing, Necdet Uzun, "IEEE 802.17 Resilient Packet Ring Tutorial," *IEEE Communications Magazine*, vol. 42, no. 3, pp. 112-118, Mar. 2004.

18. R.M. Needham and A.J. Herbert, "The Cambridge Digital Communication Ring," in *The Cambridge Distributed Computing System*, Addison-Wesley, London, UK, 1982.

19. I. Cidon and Y.Ofek, "MetaRing - A Full-Duplex Ring with Fairness and Spatial Reuse", *IEEE Trans on Communications*, vol. 41, no. 1, pp. 110-120, Jan. 1993.

20. *Token Ring Access Method and Physical Layer Specifications*, IEEE Standard 802.5, 1989.

21. F.E. Ross, "An overview of FDDI: The Fiber Distributed Data Interface," *IEEE J. on Selected Areas in Communications*, vol. 7, no. 7, pp. 1043-1051, Sept. 1989.

22. *Specification of the ATMR Protocol (v. 2.0)*, ISO Standard, IEC JTC 1/SC 6 N7873, 1993.

23. W.W. Lemppenau, H.R.van As, H.R.Schindler, "Prototyping a 2.4 Gbit/s CRMA-II Dual-Ring ATM LAN and MAN," in *Proc. of the 6th IEEE Workshop on Local and Metropolitan Area Networks*, San Diego, CA, 1993, pp. 17-18.

24. E.R. Hafner, Z. Nendal, M. Tschanz, "A Digital Loop Communication System," *IEEE Transactions on Communications*, vol. 22, no. 6, pp. 877-881, June 1974.

25. Cecil C. Reames and Ming T. Liu, "A Loop Network for Simultaneous Transmission of Variable-length Messages," *ACM SIGARCH Computer Architecture News,* vol. 3, no. 4, pp. 7-12, Dec. 1974.

26. *Scalable Coherent Interface (SCI)*, IEEE Standard 1596, 1993.

27. OPNET Technologies Inc. (2012, Nov. 18). *OPNET Modeler* [Online]. Available: http://www.opnet.com/solutions/network_rd/modeler.html

28. N. Uzun and M. Yilmaz, "Weighted Fairness," presented at the IEEE 802.17 Working Group Interim Meeting, Orlando, FL, May 2001. Available: http://grouper.ieee.org/groups/802/17/documents/presentations/may2001/nu_wf_02.pdf

29. N. Uzun and M. Yilmaz, *"Providing Enhanced Fairness,"* presented at the IEEE 802.17 Working Group Interim Meeting, Orlando, FL, May 2001. Available: http://grouper.ieee.org/groups/802/17/documents/presentations/jul2001/nu_efa_01.pdf

30. N. Uzun and M. Yilmaz, "Multi Choke Point Detection and Virtual Destination Queuing," presented at the IEEE 802.17 Working Group Interim Meeting, San Jose, CA, Sep. 2001. Available: http://grouper.ieee.org/groups/802/17/documents/presentations/sep2001/nu_mcp_03.pdf

31. V. Karighattam, N. Uzun, D. Xie, M. Yilmaz, P. Yilmaz, "Delay and Jitter Analysis for HP in the two transit buffer scheme of Darwin and Comparison," presented at the IEEE 802.17 Working Group Interim Meeting, Orlando, FL, Jan. 2002. Available: http://grouper.ieee.org/groups/802/17/documents/presentations/jan2002/vk_dwdel_02.pdf

32. N. Uzun and M. Yilmaz, "System using weighted fairness decisions in spatial reuse protocol forwarding block to determine allowed usage for servicing transmit and transit traffic in a node", U.S. Pat. #7366789, Apr. 29, 2008.

33. N. Uzun and M. Yilmaz, "System using fairness logic for mediating between traffic associated with transit and transmit buffers based on threshold values of transit buffer", U.S. Pat. # 7231471, June 12, 2007.

34. N. Uzun and M. Yilmaz, "System using weighted fairness decisions in spatial reuse protocol forwarding block to determine allowed usage for servicing transmit and transit traffic in a node", U.S. Pat. # 7016969, Mar. 21, 2006.

35. V. Gambiroza, P. Yuan, L. Balzano, Y. Liu, S. Sheafor, and E. Knightly, "Design, analysis, and implementation of dvsr: a fair high-performance protocol for packet rings," *IEEE/ACM Trans. Networking*, vol. 12, no. 1, pp. 85-102, 2004.

36. M. Yilmaz and N. Ansari, "Weighted Fairness in Resilient Packet Rings," in *Proc. of the 2007 IEEE International Conference on Communications (ICC'07)*, Glasgow, UK, June 24-28 2007, pp. 2192-2197.

37. F. Davik, A. Kvalbein, and S. Gjessing, "Performance evaluation and improvement of non-stable Resilent Packet Ring behavior," in *Proc. Part II of the 4th International Conference on Networking (ICN'05)*, ser. LNCS 3421, Reunion Island, April 17-21 2005, pp. 551-563.

38. F. Davik, A. Kvalbein, and S. Gjessing, "An Analytical Bound for Convergence of the Resilient Packet Ring Aggressive Mode Fairness Algorithm," in *Proc. 40th annual IEEE International Conference on Communications (ICC'05)*, Seoul, Korea, May 16-20 2005, pp. 281-287.

39. F. Alharbi and N. Ansari, "Distributed bandwidth allocation for resilient packet ring networks," *Computer Networks*, vol. 49, no. 2, pp. 161-171, Oct. 2005.

40. F.Alharbi and N. Ansari, "SSA: simple scheduling algorithm for resilient packet ring networks," in *IEE Proc. on Communications*, vol. 153, no. 2, pp. 183-188, Apr. 2006.

41. D. Bertsekas and R. Gallager, "Flow control," in Data Networks, 2nd Ed. Englewood Cliffs, New Jersey: Prentice-Hall, 1987, pp. 524-530.

42. M. Alvarez, E. Salami, A. Ramirez, M. Valero, "A performance characterization of high definition digital video decoding using H.264/AVC," in *Proc. IEEE International Symposium on Workload Characterization*, Austin, Texas, Oct. 2005, pp. 24-33.

43. Simula Research Laboratory. (2006, Sep. 26). *Simula RPR Simulator* [Online]. Available: http://software.simula.no/nd/rpr/

44. H. Tyan, "Design, realization and evaluation of a component-based compositional software architecture for network simulation," Ph.D. dissertation, Dept. of Electrical Eng., Ohio State Univ., 2002.

45. V. Gambiroza, P. Yuan, and E. Knightly, "The IEEE 802.17 media access protocol for high-speed metropolitan-area resilient packet rings," *IEEE Network*, vol. 18, no. 3, pp. 8-15, May-June 2004.

46. M. Yilmaz and N. Ansari, "Weighted Fairness and Correct Sizing of Secondary Transit Queue in Resilient Packet Rings", *Journal of Optical Communications and Networking*, vol. 2, no. 11, pp. 944-951, October 2010.

47. P. Setthawong and S. Tanterdtid, "Inter-ring Traffic Management in Bridged Resilient Packet Rings: Global Fairness and Buffer Overflow Prevention," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 6, no. 11, pp. 190-201, November 2006.

48. R. Pan, L. Breslau, B. Prabhakar, and S. Shenker, "Approximate fairness through differential dropping," *ACM SIGCOMM Comput. Commun Review,* vol. 33, no. 2, pp. 23-39, Apr. 2003.

49. M. Yilmaz, N. Ansari, J. H. Kao, and P. Yilmaz, "Active Queue Management for MAC Client Implementation of Resilient Packet Rings," in *Proc. of the International Conference on Comm. (ICC'09)*, Dresden, Germany, June 14-18 2009, pp. 1-5.

50. S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance", *IEEE/ACM Transaction on Networking*, vol. 1, no. 4, pp. 397-413, Aug. 1993.

51. M. Yilmaz and N. Ansari, “Achieving Destination Differentiation in Ingress Aggregated Fairness for Resilient Packet Rings by use of Weighted Destination Based Fair Dropping,” submitted to Computer Networks.

52. M. Yilmaz and N. Ansari, “Resilient Packet Rings with Heterogeneous Links,” in Proc. of the 2012 IEEE Symposium on Computers and Communications (ISCC'12), Cappadocia, Turkey, July 1-4 2012, pp. 708-712.