NAIST-IS-DT9761028

DISSERTATION

Traffic Management for the Available Bit Rate (ABR) Service in Asynchronous Transfer Mode (ATM) Networks

Manzoor Hashmani

February 08, 1999

Department of Information Systems
Graduate School of Information Science
Nara Institute of Science & Technology

Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
DOCTOR of ENGINEERING

Manzoor Hashmani

Dissertation committee: Yuji Oie, Professor
Kotaro Minato, Professor
Akira Fukuda, Professor
Hideki Sunahara, Associate Professor

Humbly Dedicated

To The Human Race

For Whose

Happiness

I Strive For

Preface

The telecommunication networks of today are evolving rapidly. The importance of telecommunications for the industry and the user has been recognized, and telecommunications is considered as one of the drivers for our new global international society. Throughout the world the politicians have taken initiatives to construct a so-called Information Highway or Information Infrastructure.

The answer to these initiatives is a broadband network with attractive services. This broadband network can transport telecommunication services, like digital TV, digital HDTV (High Definition TV), high quality videophony, high speed data transfer, video on demand, etc, which are expected to be very attractive. Industry and universities had already worked more than a decade to solve the technical problems of this broadband network. To gain experience with those new services, researchers had been experimenting with broadband networks and services since the beginning of eighties.

The ITU-T experts for broadband networks started with the definition of the broadband transfer mode. In 1988 there was only a very reduced recommendation with respect to broadband ISDN. It was already agreed then that ATM (Asynchronous Transfer Mode) would be the transfer mode for the future broadband ISDN (BISDN). Two years later in 1990, ITU-T had already prepared 13 recommendations, using the accelerated procedure. These recommendations define the basics of ATM and determine most of its parameters.

ATM transfer mode can be used for a variety of bit rate transmission, from the order of kilo bits per second up to mega and giga bits per second. ATM divides the data to be transfered into small fixed-sized packets (53 octets) called cells and those cells are statistically multiplexed onto one virtual channel (VC) / virtual path (VP). This statistical multiplexing, unpredictable statistical fluctuations of traffic flows or fault conditions within ATM network may lead to excessive cell losses or unacceptable end-to-end cell transfer delays. This situation is called as congestion. The traffic control refers to a set of actions taken by the network to avoid congestion. An additional role of traffic control is to optimize the use of network resources for the purpose of achieving realistic network efficiency.

The first objective of this dissertation is to investigate various congestion control schemes

and the issues concerning congestion control. In Chapter 2 we present a comprehensive study of the most of the congestion control schemes and the issues which must be considered when designing a new algorithm or modifying an existing algorithm. We find Enhanced Proportional Rate Control Algorithm (EPRCA) as particularly interesting due to its low implementation complexity.

The second objective of this dissertation is to propose a new or modify an exiting traffic/congestion control algorithm in order to achieve maximum throughput performance and fairness in bandwidth utilization among contending sources. In this connection we found that the Enhanced Proportional Rate Control Algorithm (EPRCA) has low implementation complexity but due to its typical handling of the high congestion and subsequent no control in no congestion it produces substantial unfairness in bandwidth utilization by the end-to-end traffic (Section 3.5). We solved this problem of EPRCA by suggesting some modifications and called new algorithm as Modified Enhanced Proportional Rate Control Algorithm (EPRCAM) which is described in Section 3.6. A comprehensive performance comparison between EPRCA and EPRCAM is also given in Chapter 3.

We investigate Explicit Rate Indication for Congestion Avoidance (ERICA) algorithm which is being extensively discussed at present. We present some of the problems that give rise to unfairness in bandwidth utilization by the contending sources which are already identified by its original proposer. We establish that the level of the unfairness due to these problems was highly underestimated. We also found new problems in ERICA which produce unfairness (see Section 4.4). Due to these problems and its high implementation complexity, we establish in Chapter 4 that ERICA is outperformed by our proposed algorithm EPRCAM. We give performance comparison of EPRCA, EPRCAM and ERICA in this chapter.

The third objective of this dissertation is to investigate robustness of EPRCAM in terms of throughput and fairness particularly when large amount of bandwidth is suddenly used or relinquished by the higher priority service classes (VBR/CBR). We find that unfortunately the throughput and fairness performance of EPRCAM gets severely affected when a large amount of bandwidth is used or relinquished by the VBR/CBR connections. Further modifications in EPRCAM are proposed in Section 5.4 in order to handle the above problem. It is shown in Section 5.6 that after proposed changes, the performance of EPRCAM is highly satisfactory against other congestion control algorithms.

We conclude this dissertation in Chapter 6 and give some guidelines for future research.

Keywords: ATM Switching, ABR Congestion Control, ABR Traffic Control, High Speed Communication, Data Communication

Acknowledgements

First of all I would like to thank Professor Yuji Oie without whose continuous help, support, guidance and encouragement, this work would not have been possible at all. He has been a constant source of inspiration for me during the course of this work. I am deeply indebted to him for all his assistance.

Special thanks go to Professor Kotaro Mianto and Professor Akira Fukuda for their help and encouragement during this research work.

I wish to express my gratitude to Associate Professor Hideki Sunahara for his help.

I am also very thankful to Assistant Professor Kenji Kawahara who selflessly guided me throughout this research work. I highly appreciate his discussions and assistance at all stages of this research. Without his help, this dissertation would not have been in its current form.

I am also thankful to Dr. Yoshihiro Nakahira who is currently working with Oki Denki Corporation of Japan for his guidance and encouragement. He has been particularly helpful as a fellow student at Nara Institute of Science & Technology.

I am thankful to Mr. Katsuyoshi Iida and other students in Oie's and Minato's Laboratory for their assiatance.

In general I am very much thankful to all Japanese people who invited me in their country to do research and helped me make my dreams come true.

I often interrupted Professor Oie's precious and even private time to write and discuss papers. Special thanks go to his family for their understanding.

I am particularly thankful to my wife, Shaheen, for being understanding during my busy schedule.

My appreciation for many friends and colleagues who provided moral and physical support behind the scene.

Finally, my greatest appreciation goes to my father and to my mother who gave me existence and without whose prayers I am nothing more than a lifeless grain of sand. May Allah (SWT) give them long life and happiness.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 B-ISDN

Up to the present, most networks are dedicated to specific purposes like telephony, TV distribution, circuit-switched or packetized data transfer [4]. Using pre-existing networks for new applications may lead to characteristic drawbacks, as such networks are not usually tailored to the needs of services that were unknown when networks were implemented. So data transfer over the telephone network is limited by a lack of bandwidth, flexibility and quality of analog voice transmission equipment. Telephone networks were engineered for a constant bandwidth service, and using them for variable bit rate data traffic requires costly adaptation.

In 1984, the Plenary Assembly of the *CCITT* (International Telegraph and Telephone Consultative Committee, which has become the International Telecommunication Union Telecommunication standardization sector, *ITU-T*, as of the beginning of 1993) adopted the I series recommendations dealing with Integrated Services Digital Network (*ISDN*) matters. The CCITT stated that an ISDN is a network, in general evolving from a telephony IDN, that provides end-to-end digital connectivity to support a wide range of services, including voice and non-voice services, to which users have access by a limit set of standard multi-purpose user-network interfaces [6]. One of ISDN standard interfaces was defined and called **basic access**, comprising two 64 kbit/s B channel and a 16 kbit/s signalling D channel. Another type of interface, the **primary rate access**, with a gross bit rate of about 1.5 Mbit/s or 2 Mbit/s, offers the flexibility to allocate high-speed H channels or mixture of B channels, H channels and a 64 kbit/s signalling channel.

This original ISDN is based on the digitized telephone network [6] which is characterized by the 64 kbit/s channel. The 64 kbit/s ISDN is basically a circuit switched network, but it can offer access to packet-switched services. ISDNs are being implemented in this decade.

Their benefits for the user and network provider include:

- common user-network interface for access to a variety of services
- enhanced (out-of-band) signalling capabilities
- service integration
- provision of new and improved services

ISDNs can offer users an interface with capacity of up to about 2 Mbit/s. However, connection of Local Area Networks (LANs) or transmission of moving images with high resolution may require considerably higher bit rate. Consequently, the conception and realization of a broadband ISDN (*B-ISDN*) was desirable.

ITU-T Recommendation I.113 [7] defines 'broadband' as 'a service or system requiring transmission channels capable of supporting rates greater than the primary rate'. Armbrüster and Rothamel [2] tried to compile technical characteristics for major B-ISDN applications. The results are given in Table 1.1.

Table 1.1: Characteristics of broadband services

Service	Bit rate (Mbit/s)	Burstiness
Data transmission	1.5 to 130	1 - 50
Document transfer / retrieval	1.5 to 45	1 - 20
Video-conference / video telephony	1.5 to 130	1 - 5
TV distribution	30 to 130	1
HDTV distribution	130	1

Burstiness = peak bit rate/average bit rate

From this table, it is found that the bit rate available to a broadband user is up to hundreds of Mbit/s. The higher bit rate on 64 kbit/s based ISDN was realized by mixing B channel, H channel and D channel. However, This could not manage a bit rate gradually existing from 64 kbit/s telephony to hundreds of Mbit/s HDTV distribution. Furthermore, channel structures are fixed at subscription time and not dynamically changed. This deadlock was finally overcome by adopting an interface model based on a complete breakdown of its payload capacity into small pieces called *cells*, each of which can serve any purpose. They may be employed to carry information relating to any type of connections: this technique is referred to as the **Asynchronous Transfer Mode** (*ATM*).

1.2 ATM

1.2.1 B-ISDN based on ATM

The asynchronous transfer mode (ATM) is considered the ground on which B-ISDN is to be built [8]. The term *transfer* comprises both transmission and switching aspects, and a transfer mode is thus a specific way of transmitting and switching information in a network.

In ATM, all information to be transferred is packed into fixed-size slots called cells. These cells have a 48 octet information field and a 5 octet header (see Fig. 1.1). Whereas the information field is available for the user, the header field carries information that pertains to the ATM layer functionality itself, mainly the identification of cells by means of a label. ATM uses a label field inside each cell header to define and recognize individual communications. In this respect, ATM resembles conventional packet transfer modes. Like packet switching techniques, ATM can provide a communication with a bit rate individually tailored to the actual need, including time-variant bit rates. In ATM based B-ISDN, a bit rate of about 155 Mbit/s or 622 Mbit/s will be typically offered to the user across the broadband user-network interface.

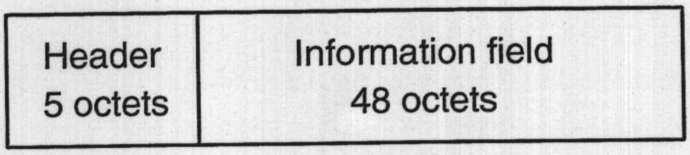


Figure 1.1: ATM cell structure

The term *asynchronous* refers to the fact that, in the context of multiplexed transmission, cells allocated to the same connection may exhibit an irregular recurrence pattern as they are filled according to the actual demand. The multiplexing and switching of cells are independent of the actual application. Thus, the same piece of equipments can handle a low bit rate connection as well as high bit rate connection, do it of stream or burst nature. Dynamic bandwidth allocation on demand with a fine degree of granularity is provided. The flexibility of the ATM-based B-ISDN network access resulting from the cell transport concept strongly supports the idea of a unique interface which can be employed by a variety of customers with quite different service needs.

ATM combines advantageous features of both connection- and packet-oriented techniques. The former requires only low overhead and processing, and, once a connection is established, the transfer delay of the information is low and constant. The latter is much more flexible in terms of the bit rate assigned to individual (virtual) connections. ATM is a connection-oriented, hardware-controlled, low-overhead concept of virtual channels which have no flow control or error recovery.

1.2.2 ATM Layer Function

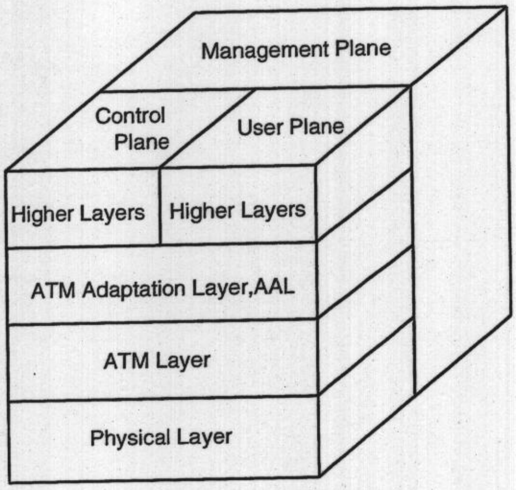


Figure 1.2: B-ISDN protocol reference model

Figure 1.2 shows the B-ISDN protocol reference model. Functions of each plane and layer in Fig. 1.2 are defined in [9]. I will only explain the function of ATM layer based on cell header description in the remainder of this section and also show that of ATM Adaptation Layer (*AAL*) in Section 1.3 in order to clarify the motivation of this dissertation.

The ATM layer is the one above the physical layer. Its characteristic features are independent of the physical medium. Figure 1.3 depicts the cell header structure. Four functions of this layer have been identified as follows [4].

- In the transmit direction, cells from individual Virtual Channel (*VC*) and Virtual Path (*VP*) are multiplexed into one resulting cell stream by the **cell multiplexing** function. The composite stream is normally a non-continuous cell flow. At the receiving side, the **cell demultiplexing** function splits the arriving cell stream into individual cell flows appropriate to the VC or VP.

The ATM layer has two hierarchical levels; virtual channel level and virtual path level, and both are defined in [7]:

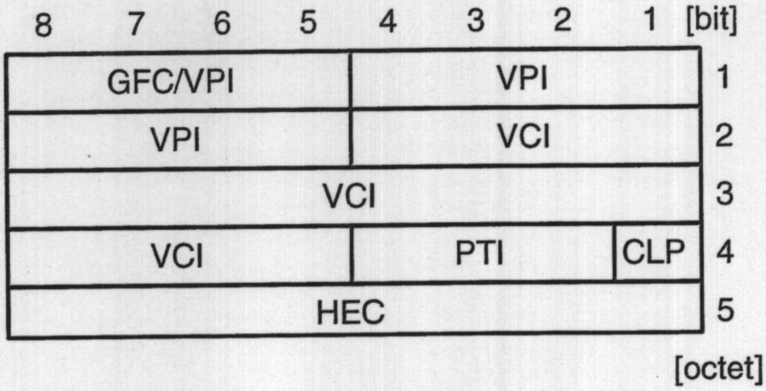


Figure 1.3: Cell header structure

Virtual channel: 'A concept used to describe unidirectional transport of ATM cells associated by a common unique identifier value.' This identifier is called the **virtual channel identifier (VCI)** and is part of the cell header. The VCI is assigned in a field of 16 bits of the cell header for routing.

Virtual path: 'A concept used to describe unidirectional transport of cells belonging to virtual channels that are associated by a common identifier value.' This identifier is called the **virtual path identifier (VPI)** and is part of the cell header. The VPI field at the B-ISDN UNI(User Network Interface) consists of 8 bits while that at NNI(Network Network Interface) comprises the first 12 bits of the cell header, thus providing enhanced routing capabilities.

A transmission path may comprise several VPs, and each VP may carry several VCs. The VP concept allows the grouping of several VCs.

- **VPI and VCI translation** are performed at ATM switching nodes and/or at cross-connect nodes. Within a VP node, the value of the VPI field of each incoming cell is translated into a new VPI value for the outgoing cell. The value of the VPI and VCI are translated into a new value at a VC switch.

- The **cell header generation/extraction** function is applied at the termination points of the ATM layer. In the transmit direction, after receiving the cell information field from the AAL, the cell header generation adds the appropriate ATM cell header except for the **Header Error Control (HEC)**. In the opposite direction, the cell header extraction function removes the cell header. Only the cell information field is passed to the AAL.

- The **GFC(Generic Flow Control)** function is only defined at the B-ISDN UNI. GFC supports control of the ATM traffic flow in a customer network. It can be used to alleviate short-term overload conditions at the UNI. The GFC field in the cell header consists of 4 bits.

Three header bits are used for the **payload type identification (PTI)**. The payload of user information cells contains service adaptation functions. The left bit of PTI is used to distinguish between user cells and F5 cells which support OAM(Operation and Management) of VCCs(Virtual Channel Connection). The center bit is allocated to the **ATM-layer-user-to-ATM-layer-user (AUU)** indication which is used by AAL type 5 (see Section 1.3). The right bit is used for the **congestion indication (CI)** bit which may be modified by any congested network element to inform the end-user about its state (see Section 1.4).

The **cell loss priority (CLP)** field consists of one bit which is used explicitly to indicate the cell loss priority. If the value of the CLP bit is '1', the cell is subject to discard, depending on the network conditions. However, the agreed quality of service (**QOS**) parameters will not be violated. In the other case (CLP = '0'), the cell has high priority and therefore sufficient network resources have to be allocated to it. The CLP bit may be set by the user or the service provider (also see Section 1.4).

The HEC field is part of the cell header, but it is not used by the ATM layer. The HEC sequence is processed by the physical layer and is specified in ITU-T Recommendation I.432 [13].

1.3 AAL : ATM Adaptation Layer

The AAL is between the ATM layer and higher layers. Its basic function is the enhanced adaptation of services provided by the ATM layers to the requirements of the higher layer [10]. AAL functions are organized in two sublayers: **segmentation and reassembly (SAR)** sublayer and **convergence sublayer (CS)**. The essential functions of the SAR sublayer are, at the transmitting side, segmentation of higher layer PDUs (Protocol Data Units) into a suitable size for the information field of the ATM cell and, at the receiving side, reassembly of the particular information fields into higher layer PDUs. The CS is service dependent.

In order to minimize the number of AAL protocols, ITU-T proposed a service classification specific to the AAL. This classification was made with respect to the following parameters:

- timing relation between source and destination (required or not required)
- bit rate (constant or variable)
- connection mode (connection-oriented or connectionless)

	Class A	Class B	Class C	Class D
Timing relation	Required		Not required	
Bit rate	Constant	Variable		
Connection mode	Connection oriented			Connectionless

Figure 1.4: Service classification for AAL

Figure 1.4 depicts the AAL classes, and several AAL protocol types are defined in following subsections. Each type consists of a specific SAR sublayer and CS. This classification fits the AAL service classes. However, no strict relationship between the AAL service classes and the AAL protocol types is requested. Other combination of the described SAR and CS protocols may be used to support specific services.

1.3.1 AAL Type 1

Constant Bit Rate (CBR) services (class A) use AAL type 1 because it receives/delivers data with a constant bit rate from/to the layer above. **Timing information** is also transferred between source and destination. Indication of lost or errored information is sent to the higher layer if these failures can not be recovered within the AAL.

SAR sublayer

The SAR-PDU consists of 48 octets and its format is shown in Fig. 1.5. The first octet includes the Protocol Control Information (PCI) and all other octets are available for the SAR-PDU payload. The PCI is subdivided into a 4 bit sequence number (SN) and a 4 bit sequence number protection (SNP) field. The SN consists of a convergence sublayer indication (C) bit and a 3 bit sequence count (SC) field. The SNP field contains a 3 bit CRC which protects the SN field and an even parity bit. The SC value of the SN makes it possible to detect the loss or misinsertion of cells.

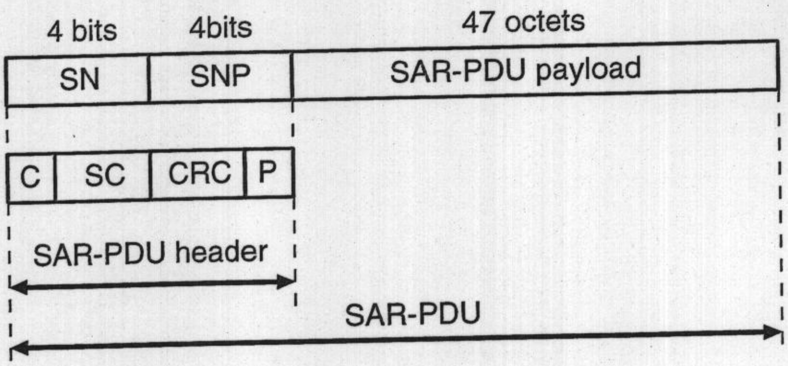


Figure 1.5: SAR-PDU format for AAL type 1

Convergence Sublayer

The function of the CS depends strongly on the service to be supported. Some of these functions are listed in the following.

- Source clock frequency recovery using the synchronous residual time stamp method (SRTS). For details of this method see [11].
- Transfer of structure information between source and destination.
- Forward Error Correction (FEC) may be used to ensure high quality for some video and audio applications. This may be combined with octet interleaving to give more secure protection against errors.

1.3.2 AAL Type 2

ATM Adaptation Layer Type 2: This AAL is still undefined by the International Standards bodies. It is a placeholder for variable bit rate video transmission.

1.3.3 AAL Type 3/4

The name of this AAL reflects its development: as the service classes were specified, separate AALs were allocated for class C and class D services, namely AAL type 3 and AAL type 4. The AAL 3 was intended to provide framing service for connection-oriented data protocols like X. 25, and the AAL 4 was intended to provide that for connectionless protocols like IP. The AAL is used for connection oriented as well as for connectionless data communication. However, the AAL itself does not perform all functions required by a connectionless service, since functions like routing and network addressing are performed on the higher layer or the network one. Therefore, both types have merged, thereby supporting both service classes.

SAR sublayer

Figure 1.6 illustrates the SAR-PDU format. In general, CS-PDUs are of variable length. When accepting such a PDU, the SAR sublayer generates SAR-PDUs containing up to 44 octets of CS-PDU data. The CS-PDU is preserved by the SAR sublayer. This requires a **segment type (ST)** indication. The ST indication identifies a SAR-PDU as being **beginning of message (BOM), continuation of message (COM), end of message (EOM)** or **single-segment message (SSM)**.

Multiplexing of multiple CS-PDUs on a single VCI/VPI is supported by a 10 bit **multiplexing identifier (MID)**. The use of the MID field allows the multiplexing of 2^{10} AAL-user-to-AAL-user connection on a single user-to-user ATM layer connection for connection-oriented data communication. For connectionless data communication, the MID field allows interleaving SAR-PDUs up to 2^{10} CS-PDUs on the same semi-permanent ATM layer VC.

Last segment and single one SAR-PDUs may contain less payload octets than the maximum of 44, and thus also need an indication of the number of valid octets in a 6 bit **length indicator (LI)** field.

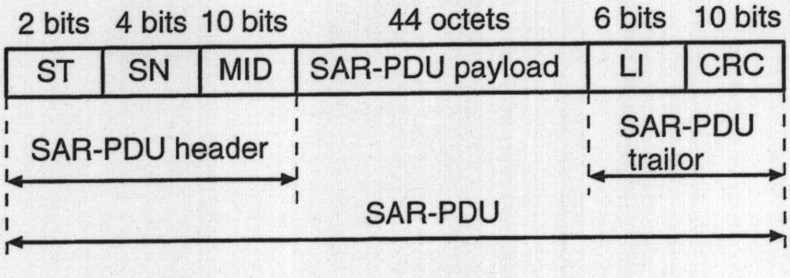


Figure 1.6: SAR-PDU format for AAL type 3/4

1.3.4 AAL Type 5

AAL type 5 will be applied to VBR sources without a timing relation between source and destination. It provides services similar to AAL type 3/4 and will mainly be used for data applications. The reason for defining this additional AAL type was its reduced overhead (i.e., SAR-PDU payload field contains only 44 octets out of a 48 octet SAR-PDU for AAL type 3/4 traffic). Its function and service mode are identical to those for AAL type 3/4. However, one essential difference is that AAL type 5 dose not support a multiplexing function, and thus there is no MID field.

SAR sublayer

The SAR sublayer accepts **service data unit (SDU)** which is of an integer multiple of 48 octets from the CS. No additional overhead is added to the received SDUs at the SAR sublayer. Only segmentation and, in the reverse direction, reassembling functions are performed. For the recognition of the beginning and end of a SAR-SDU (which is correspond to CS-PDU), AAL type 5 makes use of the AUU parameter, which is part of the PT field in the ATM header (cf. section 1.2). An AUU parameter value of '1' indicates the end of a SAR-SDU, while a value of '0' indicates the beginning or continuation of a SAR-SDU. Thus a ST field provided in AAL type 3/4 is not required.

1.4 Traffic Control and Congestion Control in ATM Networks

According to ITU-T Recommendation I.371 [12], the primary role of traffic control in B-ISDN is to protect the network and the user in order to achieve predefined network performance objectives in terms of **quality of services (QOS)** such as Cell Loss Ratio (CLR), Cell Transfer Delay (CTD) and Cell Delay Variation (CDV). Traffic control refers to the set of actions taken by the network to avoid congestion. However, congestion may occur because of misfunctioning of traffic control functions caused by unpredictable statistical fluctuations of traffic flows or of network failures, possibly leading to excessive cell losses or unacceptable end-to-end cell transfer delays. Therefore, functions referred to as congestion control ones are intended to react network congestion in order to minimize its intensity, spread and duration. Furthermore, an additional role of traffic control is to utilize network resources efficiently.

1.4.1 Traffic Control Functions

In [12], the following six functions are defined in detail as traffic control functions in ATM networks. Here, I will briefly introduce each function.

Connection Admission Control, CAC

CAC represents the set of actions taken by the network at call set-up phase in order to accept or reject an ATM connection. A connection request for a given call is accepted only when sufficient resources are available to carry the new connection through the whole network at its requested QOS while maintaining the agreed QOS of already established connections in the network.

Usage/Network Parameter Control, UPC/NPC

UPC/NPC are performed at the user-network interface (UNI) and the network-network interface (NNI), respectively, and represent the set of actions taken by the network to monitor and control traffic on an ATM connection in terms of cell traffic volume and cell routing validity. This function is also called 'police function'. The main purpose is to enforce the compliance of every ATM connection to its negotiated traffic contract.

Network Resource Management

One tool of network resource management which can be employed for traffic control is the virtual path technique. By grouping several virtual channels together into a virtual path, CAC and UPC/NPC can be simplified as only aggregated traffic of an entire virtual path has to be handled.

Priority Control

ATM cells have an explicit cell loss priority (CLP) bit in the header as previously mentioned in Section 1.2, and at least two different ATM priority classes can be distinguished. Kröner [16] describes different buffering mechanisms for switching/multiplexing systems with two cell loss priorities:

- Common buffer with pushout mechanism: Cells of both priorities share a common buffer. If the buffer is full and a high-priority cell arrives, a cell with low priority (if available) will be pushed out and lost.

- Partial buffer sharing: Low-priority cells can only access the buffer if the queue length of the buffer is less than a given threshold ($<$ total buffer capacity). High-priority cells can access the whole buffer.

- Buffer separation: Different buffers are used for the two priorities. This mechanism is simple to implement but cell sequence integrity can only be maintained if a single priority is assigned to each connection.

Traffic Shaping

Traffic shaping alters the traffic characteristics of a stream of cells on a VPC or VCC in order to reduce the peak rate, limit the burst length or reduce the cell delay variation by suitably spacing cells in time. It must maintain the cell sequence integrity of an ATM connection. It is also used in conjunction with suitable UPC functions, provided the additional delay remains within the acceptable QOS negotiated at call set-up.

Fast Resource Management

Fast resource management is a tool that enables the immediate allocation of necessary capacity, such as bit rate or buffer space, to individual burst-type connections for the duration of a burst. Indication of a burst by the user and allocation acknowledgement by the network could be signalled in-band via specific ATM layer messages.

1.4.2 ABR Traffic Control Mechanism

The **Constant Bit Rate (CBR)** and **Variable Bit Rate (VBR)** bearer capabilities have been defined by the ATM Forum for connections requiring specified ATM layer Quality of Service (QOS) commitments as regards CLR, CTD and CDV. These QOS guarantees can be delivered by the network by means of a bandwidth resource allocation process, executed at connection set-up time and denoted as the CAC function.

However, the traffic generated by data applications is highly unpredictable and of an extremely bursty nature (see Table 1.1), i.e., a highly variable packet generation rate and with varying packet sizes. It follows that the framework of a fixed static traffic contract as defined for connections using the CBR or VBR bearer capability is not ideally suited for this type of applications. Since data applications are rather sensitive to cell loss but can tolerate substantial variations in delay, there is a need for the network to inform the user via feedback of an impending exhaustion of resources which could ultimately result in cell loss. Therefore, additional ATM layer traffic management facilities are necessary in order to effectively transport traffic from such sources so that network operators can make the most of any unused link capacity without affecting the performance of the CBR and VBR connections. To this end, the *ATM Forum* and also the ITU-T are defining a novel ATM layer bearer capability, referred to as the **Available Bit Rate (ABR)** one [18]. On the other hand, the bearer capability, whose connection is not sensitive to cell loss as well as delay, is specified as **Unspecified Bit Rate (UBR)** one. ABR or UBR are usually specified in the traffic contract when the ATM network is providing **best-effort** service. Thus, these two classes of traffic are referred to as best-effort traffic [19]. Therefore, AAL type 3/4 and type 5 traffic may correspond with best-effort one.

The ATM Forum adopted a **rate based** approach [3], as opposed to a **credit based** approach with buffer allocation [17], with **closed loop** traffic control mechanism for providing an end-to-end ABR bearer capability. This ABR traffic control mechanism will allow the **Source End System (SES)** to dynamically adjust its cell sending rate based on feedback control information received from the network, indicating its availability status of bandwidth resources.

To obtain this information, the ABR connection's SES injects **Resource Management (RM)** cells into its information cell stream on a regular basis, in order to probe the network, i.e., the RM cells will actually fulfil the role of network bandwidth scouts. For an ABR traffic control mechanism that operates on a pure **end-to-end** basis, these RM cells are then returned by the **Destination End System (DES)**. Thus, for the ABR information flow from SES to DES, there exists two RM cell flows: one in the forward direction from SES to DES and another in the backward direction from DES to SES.

Moreover, the adopted rate based policy based on these RM cell flows also provides the option to safely create separately controlled ABR segments. The introduction of the concept of a **virtual destination** and of a **virtual source** allows the information control loop to be segmented at any convenient point along the end-to-end path. For any ABR connection, an intermediate network may therefore create its own independently and internally controlled segment by terminating the loop. Furthermore, it is possible to create a **link-by-link** closed loop rate control mechanism.

1.5 Overview of the Dissertation

So far, I have presented a general overview of ATM technique, for realizing broadband-ISDN(B-ISDN) in Section 1.1, especially ATM layer functions (in Section 1.2) and AAL ones which are dependent on service type such as voice on circuit emulation mode, constant and variable bit rate audio, video, data transfer and so on in Section 1.3.

Again, ATM network handles the transfer of small fixed-size cells on ATM layer. ATM enables us to transmit traffic of a wide range of bit rates, from the order of kbit/s to hundreds of Mbits/s, in unique network interface. Therefore, in ATM networks, it is expected that the statistical multiplexing of cells from different service classes of higher layer may achieve efficient utilization of network resources (i.e., bandwidth and switch buffers). On the other hand, it may cause cell loss even at any places such as UNI, NNI, intermediate ATM switch or cross-connect point along end-to-end connection. Cell loss is considered as one of main problems or drawbacks in ATM networks. Therefore, traffic control is absolutely required in ATM network to avoid congestion as presented in ITU-T Recommendation I.371 [12]. The summary of traffic control functions in given in Section 1.4.

It is considered that CAC and UPC are fundamental traffic control functions to *preventively* avoid congestion. These control functions are based on network contracts and declared parameters of traffic from users at the call set-up time. However, user traffic has highly correlated nature namely high burstiness (see Table 1.1) and may have unpredictable statistical fluctuation during established connection phase, that is, short-term congestion may still occur. In order to relieve such congestion and to minimize its intensity, spread and duration, congestion control is also recommended in [12]. Its specification is not clarified for whole AAL service class, but only for available bit rate (ABR) class, and it has been summarized in Section 1.4, in which it is required to apply not only **preventive** (or, open-loop [20, 49]) control such as CAC and UPC, but also rate-based, closed-loop **reactive** control.

In order to achieve this goal, I will concentrate on the closed-loop congestion control for ABR traffic in ATM Networks. For this purpose, I first made a comprehensive survey of most of the congestion control schemes presented till to date and identified the issues related with the congestion control (see Chapter 2). I discover that at the time of proposal of most of these schemes, it is claimed to have resolved some or all of the issues of congestion control and yet after sometime an update of that scheme or a new scheme is presented to resolve the same issues.

After careful and continuous research I find that the Enhanced Proportional Rate Control Algorithm (EPRCA) is a very interesting and promising algorithm in the terms of performance achievements and implementability. However, due to its typical handling of the high congestion, I suspect that it will have some unfairness in bandwidth allocation to the contending sources.

On further research and doing simulations of a multihop wide area network model, I discover that in EPRCA, the typical handling of high congestion state and then subsequent no control in the no congestion state resulted in substantial unfairness towards end-to-end traffic (Section 3.5). I solve this problem of EPRCA by suggesting some modifications and call the new algorithm as Modified Enhanced Proportional Rate Control Algorithm (EPRCAM) which is described in Section 3.6. A comprehensive performance comparison between EPRCA and EPRCAM is given in Chapter 3.

As the next step I investigate Explicit Rate Indication for Congestion Avoidance (ERICA) algorithm which is being extensively discussed at present. I present some of the problems that give rise to unfairness in bandwidth utilization by the contending sources which are already identified by its original proposer. However, we discovered that the level of the unfairness is highly underestimated. I also find new problems in ERICA which produce unfairness (see Section 4.4). Due to these problems and its high implementation complexity, I establish in Chapter 4 that ERICA is outperformed by our proposed algorithm EPRCAM. I also give performance comparison of EPRCA, EPRCAM and ERICA in this chapter.

Any congestion control algorithm for the ABR service has to be robust so that when large amount of bandwidth is suddenly used or relinquished by the higher priority service classes (VBR/CBR), its performance in terms of throughput and fairness must not be highly affected. After incorporating VBR/CBR functionality in our simulation program and doing further simulations, I find that unfortunately the throughput and fairness performance of EPRCAM got severely affected when a large amount of bandwidth is used or relinquished by the VBR/CBR connections. Further modifications in EPRCAM are proposed in Section 5.4 in order to handle the above problem. It is shown in Section 5.6 that after proposed changes, the performance of EPRCAM was highly satisfactory against other congestion control algorithms.

In Chapter 6, some concluding remarks and suggestions for future research are given in terms of studies in this dissertation.

Chapter 2

A Survey of Congestion Control Schemes & Issues

2.1 Congestion Control Approaches

Two congestion control approaches have been discussed: credit-based approach (open-loop control) and rate-based approach (closed-loop control).

2.1.1 Credit-Based Approach

This was the first of the two congestion control approaches which was proposed, analyzed, and implemented. The approach consists of per-link, per-VC, window flow control. Each link consists of a sender node (which can be a source end system or a switch) and a receiver node (which can be a switch or a destination end system). Each node maintains a separate queue for each VC. The receiver monitors queue lengths of each VC and determines the number of cells that the sender can transmit on that VC. This number is called "credit". The sender transmits only as many cells as allowed by the credit.

The scheme as described initially is called "Flow Controlled Virtual Circuit (FCVC)" scheme. There are two problems with this initial static version. First, if the credits are lost, the sender will not know it. Second, each VC needs to reserve the entire round trip worth of buffers even though the link is shared by many VCs. These problems were solved by introducing a credit resynchronization algorithm and an adaptive version of the scheme.

The credit resynchronization algorithm consists of both sender and receiver maintaining counts of cells sent and received for each VC and periodically exchanging these counts. The difference between the cells sent by the sender and those received by the receiver represents the number of cells lost on the link. The receiver reissues that many additional credits for that VC.

The adaptive FCVC algorithm [17] consists of giving each VC only a fraction of the round trip delay worth of buffer allocation. The fraction depends upon the rate at which the VC uses the credit. For highly active VCs, the fraction is larger while for less active VCs, the fraction is smaller. Inactive VCs get a small fixed credit. If a VC does not use its credits, its observed usage rate over a period is low and it gets smaller buffer allocation in the next cycle. The adaptive FCVC reduces the buffer requirements considerably but also introduces a ramp-up time. If a VC becomes active, it may take some time before it can use the full capacity of the link even if there are no other users.

2.1.2 Rate-Based Approach

This approach, which was eventually adopted as the standard was proposed originally by Mike Hluchyj and was extensively modified later by representatives from 22 different companies.

Original proposal consisted of a rate-based scheme of end-to-end control using a single-bit feedback from the network. In the proposal, the switches monitor their queue lengths and if congested set the EFCI bit in the cells. The destination monitors these bits for a periodic interval and if any bits are seen set, it sends an RM cell back to the source. The sources use an additive increase and multiplicative decrease algorithm to adjust their rates.

This particular algorithm uses a "negative polarity of feedback" in the sense that RM cells are sent only to decrease the rate but no RM cells are required to increase the rate. A positive polarity, on the other hand, would require sending RM cells for increase but not on decrease. If RM cells are sent for both increase and decrease, the algorithm would be called bipolar.

2.1.3 Credit-Based vs Rate-Based

After a considerable debate, which lasted for over a year, ATM forum adopted the rate-based approach and rejected credit-based approach. This debate was quite "religious" in the sense that believers of each approach had quite different goals in mind and were unwilling to compromise. To achieve their goals, they were willing to make tradeoffs that were unacceptable to the other side. Some of the key points raised during this debate are given below.

- **Per-VC Queueing**: Credit-based approach requires switches to keep a separate queue for each VC. This applies to even inactive VCs. Per-VC queueing makes switch complexity proportional to the number of VCs. Given that some large switches will support millions of VCs, this would cause considerable complexity in the switches. This was the single biggest objection to the credit-based approach and the main reason for it not being adopted. Rate-based approach does not require per-VC queueing. It can work with or without per-VC queueing. The choice is left to the implementers.

- **Zero Cell Loss**: The credit-based approach can guarantee zero cell loss under ideal conditions. Even under extreme overloads, the queue lengths cannot grow beyond the credits granted. The rate-based approach cannot guarantee cell loss. Under extreme overloads, it is possible for queues to grow large resulting in buffer overflow and cell loss. The rate-based camp considered the loss acceptable arguing that with large buffers, the probability of loss is small. Also, they argued that in reality there is always some loss due to errors and, therefore, the user has to worry about loss even if there is zero congestion loss.

- **Ramp-up Time**: The static credit-based approach allows VCs to ramp up to the full rate very fast. In fact, any free capacity can be used immediately. Some rate-based schemes and the adaptive credit-based approach can take several round trip delays to ramp up.

- **Isolation and Misbehaving Users**: A side benefit of the per-VC queueing is that misbehaving users cannot disrupt the operation of well-behaving users. However, this is less true for the adaptive scheme than for the static credit scheme. In the adaptive scheme, a misbehaving user can get a higher share of buffers by increasing its rate. Note that isolation is attained by per-VC queueing and not so much by credits. Thus, if required, a rate-based switch can also achieve isolation by implementing per-VC queueing.

- **Buffer Requirements**: The buffer requirements for the credit-based schemes were found to be less than those in the rate-based scheme with binary feedback. However, this disadvantage disappeared when explicit rate schemes were added. In credit-based approach, per-VC buffer requirement is proportional to link delay, while in the rate-based approach, total buffer requirement is proportional to the end-to-end delay. Note that the queueing delays have to be added in both cases since it delays the feedback and adds to the reaction time.

- **Delay Estimate**: Setting the congestion control parameters in the credit-based approach requires knowledge of link round trip delay. At least, the link length and speed must be known. This knowledge is not required for rate-based approaches (although it may be helpful).

- **Switch Design Flexibility**: The explicit rate schemes provide considerable flexibility to switches in deciding how to allocate their resources. Different switches can use different mechanisms and still interoperate in the same network. For example, some switches can opt for minimizing their queue length, while the others can optimize their throughput, while still others can optimize their profits. On the other hand, the credit-based approach dictated that each switch use per-VC queueing with round-robin service.

- **Switch vs End-System Complexity**: The credit-based approach introduces complexity in the switches but may have made the end-system's job a bit simpler. The proponents of credit-based approach argued that their host network interface card (NIC) is much simpler since they do not schedule each and every cell. As long as credits are available, the cells can be sent at the peak rate. The proponents of the rate-based approach countered that NIC cards have to have schedulers for their CBR and VBR traffic and using the same mechanism for ABR does not introduce too much complexity.

  The rate-based approach won by a vote of 7 to 104.

2.2 Congestion Control Schemes

Various congestion control schemes have been proposed till today. In the following sections I describe some of the prominent ones.

2.2.1 Fast Resource Management

This proposal from France Telecom [21], requires sources to send a resource management (RM) cell requesting the desired bandwidth before actually sending the cells. If a switch can not grant the request it simply drops the RM cell. After the time out, source resends the request. If a switch can satisfy the request, it passes the RM cell on to the next switch. Finally, the destination returns the cell back to the source which can then transmit the burst.

As described above, the burst has to wait for at least one round trip delay at the source even if the network is idle (as is often the case). To avoid this delay, an "immediate transmission (IT)" mode was also proposed in which the burst is transmitted immediately following the RM cell. If a switch can not satisfy the request, it drops the cell and the burst and sends an indication to the source.

If cell loss, rather than bandwidth is of concern, the resource request could contain the burst size. A switch would accept the request only if it had that many buffers available.

The fast resource management proposal was not accepted at the ATM Forum primarily because it would either cause excessive delay during normal operation or excessive loss during congestion.

2.2.2 Delay-Based Rate Control

This proposal made by Fujitsu [22] requires that the source monitor the round trip delay by periodically sending resource management (RM) cells that contain time stamp. The cells are returned by the destination. The source uses the time stamp to measure the round trip delay and to deduce the level of congestion. This approach which is similar to that described in [23], has the advantage that no explicit feedback is expected from the network and, therefore, it will work even if the path contained non-ATM networks.

Although the proposal was presented at the ATM Forum, it was not followed up and the precise details of how the delay will be used were not presented. Also, this method does not really require any standardization, since any source-destination pair can do this without involving the network.

2.2.3 Backward Explicit Congestion Notification (BECN)

This method presented in [24, 25, 26] and consists of switches monitoring their queue length and sending an RM cell back to source if congested. The sources reduce their rates by half on the receipt of the RM cell. If no BECN cells are received within a recovery period, the rate fro that VC is doubled once each period until it reaches the peak rate. To achieve fairness, the source recovery period was made proportional to the VC's rate so that lower the transmission rate the shorter the source recovery period.

This scheme was dropped because it was found to be unfair. The sources receiving BECNs were not always the ones causing congestion [27].

2.2.4 Early Packet Discard (EPD)

This method presented by Sum Microsystems [28] is based on the observation that a packet consists of several cells. It is better to drop all cells of one packet then to randomly drop cells belonging to different packets. The method uses a bit in the cell header to indicate "end of message (EOM)". ATM adaptation layer 5 (AAL5) does provide this bit. When a switch's queues start getting full, it looks for the EOM marker and it drops all future cells of the VC until the EOM marker is seen again.

It was pointed out [29] that the method may not be fair in the sense that the cell to arrive at a full buffer may not belong to the VC causing congestion.

Note that this method does not require any inter-switch or source-switch communication and, therefore, it can be used without any standardization. Many switch vendors are implementing it.

2.2.5 Link Window with End-to-End Binary Rate

This method presented by Tzeng and Siu [32], consisted of combining good features of the credit-based and rate-based proposals being discussed at the time. It consists of using window flow control on every link and to use binary (EFCI-based) end-to-end rate control. The window control is per-link (and not per-VC as in credit-based scheme). It is, therefore, scalable in terms of number of VCs and guarantees zero cell loss. Unfortunately, neither the credit-based nor the rate-based camp found it acceptable since it contained elements from the opposite camp.

2.2.6 Fair queueing with Rate and Buffer feedback

This proposal from Xerox and Cisco [33] consists of sources periodically sending RM cells to determine the bandwidth and buffer usage at their bottlenecks. The switches compute fair share of VCs. The minimum of the share at this switch and that from previous switches is placed in the RM cells. The switches also monitor each VC's queue length. The maximum of queue length at this switch and those from previous switches is placed in the same RM cell. Each switch implements fair queueing, which consists of maintaining a separate queue for each VC and computing the time at which the cell would finish transmission if the queue were to be served round-robin one-bit at a time. The cells are scheduled to transmit in this computed time order.

The fair share of a VC is determined as the inverse of the interval between the cell arrival and its transmission. The interval reflects the number of other VCs that are active. Since the number and hence the interval is random, it was recommended that the average of several observed intervals be used.

2.2.7 Proportional Rate Control Algorithm (PRCA)

This algorithm presented by Mike Hluchyj solved the "negative polarity of feedback" problem of its preceding version. The problem was solved by "positive polarity of feedback". The sources set EFCI bits on every cell except the nth cell. The destination will send an "increase" RM cell to source if they receive and cells with the EFCI bit clear. The sources keep decreasing their rate until they receive a positive feedback. Since the sources decrease their rate proportional to the current rate, this scheme was called "proportional rate control algorithm (PRCA) [34]"

PRCA was found to have a fairness problem. Given the same level of congestion, at all switches, the VCs travelling more hops have a higher probability of having their bit set than those travelling smaller number of hops. Thus long path VCs have a very few opportunities to increase and are beaten down more often than short path VCs. This was called the "beat-down problem [35]".

2.2.8 The MIT Scheme

In July 1994, Jain [36] argued that the binary feedback was too slow for rate-based control in high-speed networks and that an explicit rate indication would not only be faster but would offer more flexibility to switch designers.

The single-bit binary feedback can only tell the source whether it should go up or down. It was designed in 1986 for connectionless networks in which the intermediate nodes had no knowledge of flows or their demands. The ATM networks are connection oriented. The switches know exactly who is using the resources and the flow paths are rather static. This increased information is not used by the binary feedback scheme.

Secondly and more importantly, the binary feedback schemes were designed for window-based controls and are too slow for rate-based controls. With window-based control a slight difference between the current window and the optimal window will show up as a slight increase in queue length. With rate-based control, on the other hand, a slight difference in current rate and the optimal rate will show up as continuously increasing queue length [15, 37]. The reaction times have to be fast. We can no longer afford to take several round trips that the binary feedback requires to settle to the optimal operation. The explicit rate feedback can get the source to the optimal operating point within a few round trips.

The explicit rate schemes have several additional advantages. First, policing is straight forward. The entry switches can monitor the returning RM cells and use the rate directly in their policing algorithm. Second with fast convergence time, the system come to the optimal operating point quickly. Initial rate has less impact. Third, the schemes are robust against errors in or loss of RM cells. The next correct RM cell will bring the system to the correct operating point.

Jain substantiated his arguments with simulation results for an explicit rate scheme designed by Anna Charny during her master thesis work at the Massachusetts Institute of

Technology (MIT) [36]. The MIT scheme consists of each source sending an RM cell every nth data cell. The RM cell contains the VC's current cell rate (CCR) and a "desired rate." The switches monitor all VC's rates and compute a "fair share." Any VC's whose desired rate is less than the fair share is granted the desired rate. If a VC's desired rate is more than the fair share, the desired rate field is reduced to the fair share and a "reduced bit" is set in the RM cell. The destination returns the RM cell back to the source, which then adjusts its rate to that indicated in the RM cell. If the reduced bit is clear, the source could demand a higher desired rate in the next RM cell. If the bit is set, the source use the current rate as the desired rate in the next RM cell.

Charny also showed that the MIT scheme achieve max-min optimality in $4k$ round trips, where k is the number of bottlenecks.

This proposal was well received except that the computation of fair share requires order n operations, where n is the number of VCs. Search for an O(1) scheme led to the EPRCA algorithm discussed next.

2.2.9 Enhanced Proportional Rate Control Algorithm (EPRCA)

The merger of PRCA with explicit rate scheme lead to the "Enhanced PRCA (EPRCA)" scheme at the end of July 1994 ATM Forum meeting [59, 61]. In EPRCA, the sources send data cells with EFCI set to 0. After every n data cells, they send an RM cell. The RM cells contain desired explicit rate (ER), current cell rate (CCR), and a congestion indication (CI) bit. The sources initialize the ER field to their peak cell rate (PCR) and set the CI bit to zero.

The switches compute a fair share and reduce the ER field in the returning RM cells to the fair share if necessary. Using exponential weighted averaging a mean allowed cell rate (MACR) is computed and the fair share is set at a fraction of this average:

```
MACR  = (1-a) * MACR  + a * CCR


   Fair Share  = DPF * MACR
```

Here, a is the exponential averaging factor and DPF is a multiplier (called down pressure factor) set close to but below 1. The suggested values of a and DPF are 1/16 and 7/8, respectively.

The destinations monitor the EFCI bits in data cells. If the last seen data cell had EFCI bit set, they mark the CI bit in the RM cell.

In addition to setting the explicit rate, the switches can also set the CI bit in the returning RM cells if their queue length is more than a certain threshold.

The sources decrease their rates continuously after every cell.

```
ACR = ACR * RDF
```

Here, RDF is the reduction factor. When a source receives the returned RM cell, it increases its rate by an amount AIR if permitted.

```
IF CI=0 Then New ACR = Min(ACR+AIR, ER, PCR)
```

If CI bit is set, the ACR is not changed.

Notice that EPRCA allows both binary-feedback switches and the explicit feedback switches on the path. The main problem with EPRCA as described here is the switch congestion detection algorithm. It is based on queue length threshold. If the queue length exceeds a certain threshold, the switch is said to be congested. If it exceed another higher threshold, it said to be very highly congested. This method of congestion detection was shown to result in unfairness. Sources that start up late were found to get lower throughput than those which start early.

The problem was fixed by changing to queue growth rate as the load indicator. The change in the queue length is noted down after processing, say, K cells. The overload is indicated if the queue length increases [30, 31].

2.2.10 OSU Time-based Congestion Avoidance

Jain, Kalyanaraman, and Viswanathan at the Ohio State University (OSU) have developed a series of explicit rate congestion avoidance schemes. The first scheme [38, 39] called the OSU scheme consists of switches measuring their input rate over a fixed "averaging interval" and comparing it with their target rate to compute the current Load factor z:

```
                          Input rate
Load Factor    z = -------------
                          Target Rate
```

The target rate is set at slightly below, say, 85-95bandwidth. Unless the load factor is close to 1, all VCs are asked to change (divide) their load by this factor z. For example, if the load factor is 0.5, all VCs are asked to divide their rate by a factor of 0.5, that is, double their rates. On the other hand, if the load factor is 2, all VCs would be asked to halve their rates.

Note that no selective feedback is taken when the switch is either highly overloaded or highly underloaded. However, if the load factor is close to one, between $1 - \delta$ and $1 + \delta$

for a small δ, the switch gives different feedback to underloading sources and overloading sources. A fair share is computed as follows:

```
                         Target Rate
Fair Share = ------------------------
                  Number of Active Sources
```

All sources, whose rate is more than the fair share are asked to divide their rates by $z = (1 + \delta)$ while those below the fair share are asked to divide their rates by $z = (1 - \delta)$. This algorithm called "Target Utilization Band (TUB) algorithm" was claimed to lead to fairness [38].

The OSU scheme has three distinguishing features. First, it is a congestion avoidance scheme. It gives high throughput and low delay. By keeping the target rate slightly below the capacity, the algorithm ensures that the queues are very small, typically close to 1, resulting in low delay. Second, the switches have very few parameters compared to EPRCA and are easy to set. Third, the time to reach the steady state is very small. The source reach their final operating point 10 to 20 times faster than that with EPRCA.

In the original OSU scheme, the sources were required to send RM cells periodically at fixed time interval. This meant that the RM cell overhead per source was fixed and increased as the number of sources increases. This was found to be unacceptable leading to the count-based scheme described next.

In the count-based scheme [38], the sources send RM cells after every n data cells, as in EPRCA. The switch rate adjustment algorithm is changed to encourage quick rise. Sources below the fair share are asked to come up to the fair share regardless of the load level and those above the fair share are asked to adjust their rates by the load factor. This allows the scheme to keep the three distinguishing feature while making the overhead independent of number of VCs. Newer versions of the OSU scheme, named "ERICA" (Explicit Congestion Indication for Congestion Avoidance) and "ERICA+" are count-based [41, 42].

2.2.11 Congestion Avoidance using Proportional Control (CAPC)

Andy Barnhart from Hughes Systems has proposed a scheme called "Congestion Avoidance using Proportional Control (CAPC) [43]". In this scheme, as in OSU scheme, the switches set a target utilization slightly below 1. This helps keep the queue length small. The switches measure the input rate and load factor z, as in OSU scheme, and use it to update the fair share.

During underload ($z < 1$), fair share is increased as follows:

```
Fair share = Fair share * Min(ERU, 1 + (1 - z) * Rup)
```

Here, *Rup* is a slope parameter in the range 0.025 to 0.1. ERU is the maximum increase allowed and was set to 1.5.

During overload ($z > 1$), fair share is decreased as follows:

```
Fair share = Fair share * Max(ERF, 1 - (z - 1) * Rdn)
```

Here, *Rdn* is a slope parameter in the range 0.2 to 0.8 and ERF is the minimum decrease required and was set to 0.5.

The fair share is the maximum rate that the switch will grant to any VC.

In addition to the load factor, the scheme also uses a queue threshold. Whenever the queue length is over this threshold, a congestion indication (CI) bit is set in all RM cells. This prevents all sources from increasing their rate and allows the queues to drain out.

The distinguishing feature of CAPC is oscillation-free steady state performance. The frequency of oscillations is a function of 1 - z, where z is the load factor. In steady state, z = 1, the frequency is zero, that is, the period of oscillations is infinite.

2.2.12 Explicit Rate Indication for Congestion Avoidance (ERICA)

ERICA [53, 54, 55, 56] is a congestion avoidance algorithm for ABR traffic which computes explicit rate to be indicated to sources in RM cells. ERICA consists of a few options besides the following basic steps.

An ERICA switch periodically monitors load on each link and determines a load factor, z, the available capacity, and the number of currently active VCs. The load factor is calculated as the ratio of the measured input rate at the port to the target capacity of the output link:

$$z = (\text{ABR Input Rate})/(\text{ABR Capacity}),$$

where: ABR Capacity = Target Utilization x Link Bandwidth - VBR Usage - CBR Usage

The input rate is measured over an interval called the *switch measurement interval*. In basic ERICA algorithm, the target utilization is set to a fraction (close to, but less than 100%) of the available capacity. However, in ERICA+, the target utilization is dynamically changed depending on the filling of switch queues. It is the ERICA+ that we are targeting in our paper so that we shall denote ERICA+ as ERICA for simplicity in this paper.

The load factor, z, is an indicator of the congestion level of the link. The optimal operating point is at a load factor value of one.

The fair share of each VC, *FairShare*, is computed as follows:

FairShare=(ABR Capacity)/(Number of Active Connections).

Each connection can send at a rate up to *FairShare*. If a connection does not use all of its *FairShare*, then the switch fairly allocates the remaining capacity to other connections using a quantity called *VCShare* computed as follows:

VCShare=CCR/z.

The explicit rate (Explicit_Rate) is calculated as:

Explicit_Rate=Max(FairShare,VCShare).

If the *VCShare* value is greater than *FairShare* value, the source is allowed to send at *VCShare*.

2.3 Congestion Control Issues

2.3.1 Scalability

Networks are generally classified based on extent (coverage), number of nodes, speed, or number of users. Since ATM networks are intended to cover a wide range along all these dimensions, it is necessary that the scheme be not limited to a particular range of speed, distance, number of switches, or number of VCs. In particular, this ensures that the same scheme can be used for local area networks (LANs) as well as wide area networks (WANs).

2.3.2 Optimality

In a shared environment the throughput for a source depends upon the demands by other sources. The most commonly used criterion for what is the correct share of bandwidth for a source in a network environment, is the so called "max-min allocation [14]." It provides the maximum possible bandwidth to the source receiving the least among all contending sources. Mathematically, it is defined as follows. Given a configuration with n contending sources, suppose the ith source gets a bandwidth x_i. The allocation vector $x_1, x_2, ..., x_n$ is feasible if all link load levels are less than or equal to 100infinite. For each allocation vector, the source that is getting the least allocation is in some sense, the "unhappiest source." Given the set of all feasible vectors, find the vector that gives the maximum allocation to this unhappiest source. Actually, the number of such vectors is also infinite although we have narrowed down the search region considerably. Now we take this "unhappiest source" out and reduce the problem to that of remaining $n-1$ sources operating on a network with reduced link capacities. Again, we find the unhappiest source among these $n-1$ sources, give that source the maximum allocation and reduce the problem by one source. We keep repeating this process until all sources have been given the maximum that they could get.

2.3.3 Robustness

The scheme should be insensitive to minor deviations. For example, slight mistuning of parameters or loss of control messages should not bring the network down. It should be possible to isolate misbehaving users and protect other users from them.

2.3.4 Implementability

The scheme should not dictate a particular switch architecture. This is a very important point in final selection since many schemes are found to be too complicated to be implemented.

Chapter 3

Proposal of Modified EPRCA (EPRCAM)

3.1 Introduction

The primary role of traffic control in ATM networks is to protect the network and the user in order to achieve predefined network performance objectives in terms of cell loss ratio, cell transfer delays and cell delay variance [12]. Basically, traffic control refers to a set of actions by the network to avoid and relieve congestion. The congestion of long duration may result in cell loss, poor throughput and underutilization of network [49]. Hence, it is highly desirable to relieve the network of congestion as soon as possible.

The congestion relief, in turn, depends on congestion control scheme. The performance of a network in terms of throughput and fairness in bandwidth allocation among contending sources depends on the congestion control or the congestion avoidance algorithm employed. I investigate *Enhanced Proportional Rate Control Algorithm (EPRCA)* which is one of the candidates to be employed in ATM networks and is recommended by the ATM forum as one of the explicit rate control algorithms in [48]. I show that EPRCA has certain deficiencies which result in unfairness in bandwidth utilization. We suggest some modification in EPRCA and call it *Modified Enhanced Proportional Rate Control Algorithm (EPRCAM)*. I show through arguments that EPRCAM leads to better performance both in terms of throughput and fairness in utilization of the available bandwidth by the ABR sources. To substantiate our claims I give a performance comparison between EPRCA and EPRCAM using computer simulations of a multihop wide area network (WAN).

I show that EPRCAM gives 5-10% better throughput performance than that of EPRCA. However, the most important achievement of EPRCAM is its fairness in bandwidth utilization which is around 30% improvement over the EPRCA.

The rest of the chapter is structured as follows. After explaining basic mechanism of congestion control in section 3.2, we describe the EPRCA algorithm in section 3.3. The simulation environment is explained in section 3.4. Section 3.5 identifies the deficiencies in EPRCA and section 3.6 explains our proposed modification. Section 3.7 gives performance comparison of EPRCA and EPRCAM algorithms through simulation results. Lastly, I conclude in section 3.8.

3.2 Congestion Control: Basic Mechanism

Before explaining problems of EPRCA, I describe the basic mechanism of congestion control and some terminology defined in [48] and used here.

- After every N_{rm}-1 (= 31) data cells or a fixed time interval Trm (= 100 milliseconds), a source sends a resource management (RM) cell which travels from the source to destination and back to the source carrying congestion information of the network in it. In its forward direction, it is called *forward* RM (FRM) cell and in the backward direction as *backward* RM cell (BRM).
- A switch may reduce the explicit rate (ER) in FRM and/or BRM cells to the rate it can support (called fairshare) for the source to which the RM cell belongs.
- A switch may also set the congestion indication (CI) bit in the RM cell.
- The calculation of the fairshare and CI bit is done in accordance with some congestion control mechanism.

3.3 Enhanced Proportional Rate Control Algorithm (EPRCA)

3.3.1 Switch Behavior

In EPRCA algorithm a switch maintains three congestion states namely *no congestion* (NCONG), *low congestion* (LCONG) and *high congestion* (HCONG) state. The congestion states are decided according to the following rules depending upon two thresholds on the length of queue in the switch (SWQ). They are, a *low threshold* (LTH) and a *high threshold* (HTH).

```
IF SWQ>=HTH THEN HCONG=TRUE
IF SWQ>=LTH AND SWQ<HTH THEN LCONG=TRUE
```

A switch also maintains an exponential average of CCR (Current Cell Rate) values of sources taken from the FRM cells of the virtual connections (VCs) passing through it. This average is called Mean of Allowed Cell Rates (MACR) and is calculated according to the following expression.

```
MACR=(1-a)*MACR+a*CCR
```

where, the recommended value of "a" is 1/16 [48].

On arrival of a BRM cell, if a switch is either in a low or a high congestion state, it calculates *explicit rate* (ER) for that VC using the MACR value.

The behavior of a switch employing EPRCA algorithm mainly consists of the following rules.

```
IF (HCONG) AND (ER>=MACR*MRF) THEN
        CI = 1
        ER = Min(ER,MACR*MRF)
ELSE IF (LCONG) AND (ER>=MACR*DPF)
        CI = 1
        ER = Min(ER,MACR*ERF)
END (*IF*)
```

The recommended values of *explicit reduction factor* (ERF) and *major reduction factor* (MRF) are 15/16 and 1/4 respectively [48].

The above algorithm means that the switch provides no response in the no congestion state while in low congestion and high congestion states a switch notifies its sources to reduce their rates by a small and a very large fraction respectively.

3.3.2 Source Behavior

When a source receives a BRM cell, it adjusts its rate of transmission according to the following rules.

1. If no congestion is indicated by the switch, then allowed cell rate (ACR) is increased by PCR*RIF. The recommended value of RIF (rate increase factor) is 1/16 [48].
2. If congestion is indicated by the switch, ACR is reduced by ACR*RDF and minimum of ACR and ER is selected. The recommended value of RDF (rate decrease factor) is 1/16 [48].

3.4 Network Simulation Model

3.4.1 Description of Model & Parameter Setting

As depicted in Figure 3.1, our network simulation model is a multistage wide area network with four switches (4x4, output buffer type), thirteen sources and three destinations with each link of 120Km. The names of output buffers in each switch are Q0, Q1, Q2 and Q3. The sources are divided into four groups. The dashed lines show the traffic flow. The purpose of choosing such a complex model is to make it as realistic as possible.

Group-A: Source T0 to Source T3 (end-to-end traffic)
Group-B: Source T4 to Source T6 (two-hop traffic)
Group-C: Source T7 to Source T9 (two-hop traffic)
Group-D: Source T10 to Source T12 (one-hop traffic)

The values of parameters used are given below. Other parameters are taken as recommended by ATM Forum [48].

- low threshold: 20 cells, high threshold: 100 cells.
- link capacity = 155.52 Mbps

3.4.2 Fairness Definition

Since, a strict definition of unfairness is not available, I give my own definition of unfairness as follows.

$$F_n(\%) = (T_n - T_i)/T_i * 100$$

such that $T_i \leq T_j$ For $j = 0, 1, \ldots.., N$
where, F_n = Unfairness in % for source n,
and, T_n = Normalized actual throughput of source n.

In the above expression, $F_n(\%)$ gives the difference between the throughput of a source and that of the source achieving minimum throughput. The condition in the "where" clause states that the fairness comparison can only be done between the sources which expect the same throughput (according to max-min criteria in [48] page 90).

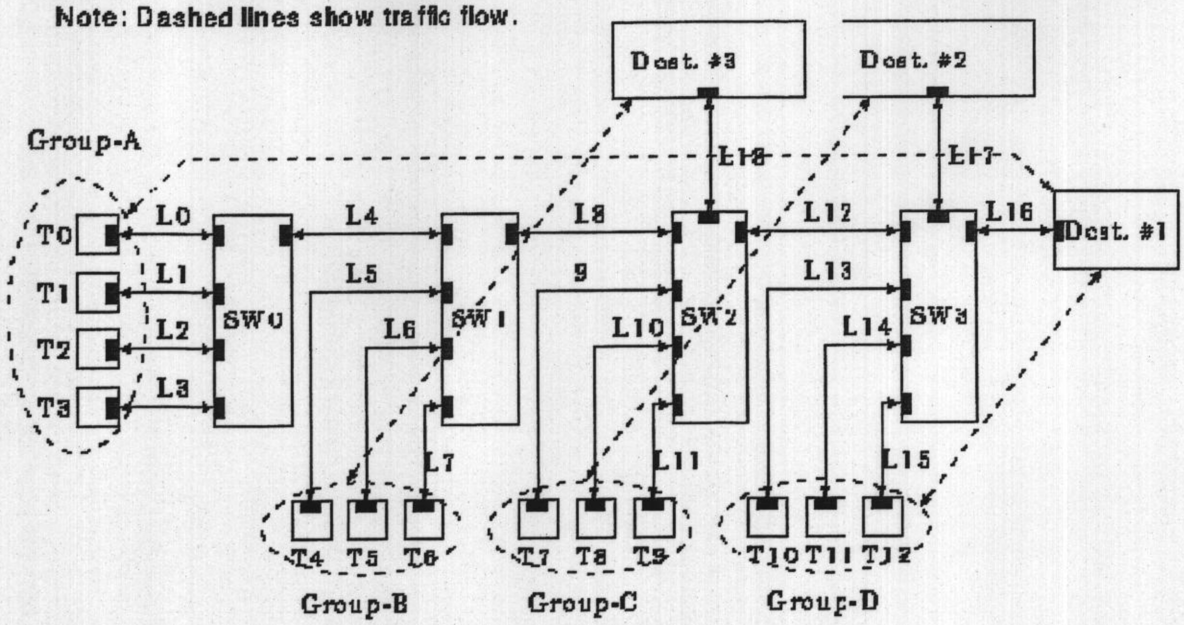


Figure 3.1: Network Simulation Model

3.4.3 Description of Figures and Tables

Below is a description of the figures and tables given in this chapter. Some of the captions described below mention EPRCAM which is defined in section 3.6.

Table 3.1 summarizes throughput and unfairness results of all sources and some links (L8, L12 and L16) for EPRCA and EPRCAM algorithms when the PCR value of all the sources is 30Mbps. The corresponding results when PCR value of all the sources is 60Mbps are shown in Table 3.2.

Table 3.3 gives throughput and unfairness results for EPRCAM algorithm for different values of RIFS. The RIFS is a new parameter introduced in the EPRCAM algorithm.

Looking at the model depicted in Figure 3.1, the traffic pattern is selected in such a way that seven VCs pass through each of the link L8, L12 and L16.

Figures 3.2a & 3.2b show rates of Group-A sources for EPRCA and EPRCAM respectively. The rates are given in the unit of cells per second. However, Figures 3.3a & 3.3b show throughput curves of only Group-D sources for EPRCA and EPRCAM algorithm respectively. Here the PCR value of all sources is 30 Mbps except T10 whose PCR value is 60 Mbps.

Figures 3.4a & 3.4b refer to EPRCA and EPRCAM scheme respectively. Each figure shows source transmission rates of T0 (end-to-end traffic) & T4 (cross traffic) for PCR values of 30 Mbps. The corresponding results for PCR values of 60 Mbps are shown in in Figures 3.5a & 3.5b. Figures 3.4 & 3.5 correspond to only a portion of simulation time which is from time slot 110,000 to time slot 120,000.

3.5 Problem Identification

Now, I describe and identify the problems with the EPRCA and their possible impact on the throughput and fairness performance of a network.

3.5.1 High congestion state problem

If the high congestion state is brief which is more likely due to major reductions in source rates, then there is a possibility that only some unlucky sources whose RM cells pass through the switch in that state and not all of the sources are notified to reduce their rates. Since, a high congestion state is usually followed by a no congestion state, the sources which are not notified during high congestion state to reduce their rates, start recovery from much higher rates, which leads to substantial unfairness.

3.5.2 No-Control-In-No-Congestion-State problem

Most likely the high congestion state will be followed by a no congestion state with switch queues almost empty due to substantial reduction in source rates, suggesting underutilization of the network. In the no congestion state the EPRCA algorithm does not provide explicit rate notification and relies on sources to gradually increase their rates [48][49] which leads to underutilization of the network resources.

3.5.3 PCR dependent rate increment problem

According to current recommendation of ATM Forum for EPRCA [48] (section 3.3), all sources should increase their rates by a factor of their PCRs when there is no congestion at the switch. Obviously those sources whose PCRs are high will get bigger chunk of the bandwidth, thereby causing unfairness.

In brief, the major reduction done in source rates during high congestion state and then subsequent no control exerted by the switch in the no congestion state cause unfairness and underutilization.

3.6 Proposed Solution: EPRCAM

3.6.1 Modified EPRCA

It is already an established fact that switch queues below a threshold indicate underload. In case of EPRCA, the sources are allowed to increase their rates by a factor of PCR in a situation of underload. As a result, sources may increase their rates from different points and possibly with different amount of increment depending on their PCR value. I suggest that for synchronization and fairness purposes, even when the switch queues are below a threshold indicating no congestion, it should be the switch and not sources which should control the increment in the source rates.

Hence, I propose that in no congestion state as well, a switch should calculate an explicit rate for sources passing through it. The rate can be calculated by using MACR value according to the following expression.

```
        ER=Min(ER,MACR*(1+RIFS))
```

where, $RIFS$ = Rate Increase Factor for Switch.

I call the proposed algorithm as Modified EPRCA (EPRCAM). Here is the switch and source behavior in EPRCAM algorithm.

3.6.2 Switch Behavior

The Mean of Allowed Cell Rates is calculated as follows.

```
        MACR=(1-a)*MACR+a*CCR
```

where, the recommended value of "a" is 1/16.

```
IF (HCONG) AND (ER>=MACR*MRF) THEN
        ER = Min(ER,MACR*MRF)
END (*IF*)

IF (LCONG) AND (ER>=MACR*DPF) THEN
        ER = Min(ER,MACR*ERF)
END (*IF*)

IF (NCONG) THEN ER = Min(ER,MACR*(1+RIFS))
```

3.6.3 Source Behavior

The source behavior is the same as described in section 3.3.2. However, the RIF is set to 1 in order for the source to always set its ACR to the ER value in the returning RM cells.

3.6.4 Advantages of EPRCAM

- *Unfairness due to high congestion problem eliminated:* The unfairness caused due to the high congestion state problem described in section 3.5 will be minimized due to our proposed modification. This is because usually a high congestion state is followed by a no congestion state and in the no congestion state all source rates are synchronized according to the optimal operating point of a network (MACR) even if some of the sources start recovery from rates which are widely apart.

- *Unfairness due to PCR dependent increment eliminated*: The unfairness caused due to the dependence on PCR values of sources for increment in no congestion state has been effectively eliminated due to the fact that the sources are explicitly notified of transmission rates even in the no congestion state by the switch.

3.6.5 Impact of RIFS

The RIFS [1] parameter is used to compute increment when a switch is in no congestion state. Since, I know that end-to-end traffic is slower to respond to changes due to longer round trip propagation delay than the cross traffic, I expect that smaller values of RIFS will produce better fairness in bandwidth utilization on the cost of some degradation in bandwidth utilization.

The impact of RIFS is examined through simulations in section 3.7.4.

3.7 Performance Comparison (EPRCAM v/s EPRCA)

3.7.1 Unfairness due to high congestion problem eliminated

Due to the high initial cell rate values and no control exerted by the network for the first round trip after the simulation is started, switches definitely go into high congestion state. This gives us an opportunity to observe the behavior of EPRCA and EPRCAM in terms of the problems discussed in sections 3.5.1 and 3.5.2. This situation is depicted in Figure 3.2a, in which due to a high congestion and the subsequent no congestion state, there is

---

[1] There is no extra hardware requirement for EPRCAM.

a huge gap in recovery points between source T2 and the rest of Group-A sources. For the remaining period of no congestion, all sources are recovering slowly. However, T2 is recovering from a much lower rate than other sources, which causes very high unfairness. Obviously this situation did not occur in EPRCAM, because of explicit notification of rates in all states due to which source rate become synchronized even if they start recovery from different rates.

3.7.2 Unfairness due to PCR dependent increment eliminated

A simulation was run on network model described earlier with PCR of each source set to 30 Mbps except for source T12 whose PCR is set to 60Mbps. The PCR of T12 is set higher to underline unfairness caused by different PCR values. Figures 3.2a & 3.2b show source rates (ACR) of Group-A sources for EPRCA and EPRCAM algorithm respectively. Whereas, Figures 3.3a & 3.3b show throughput curves of Group-D sources.

We can clearly see the impact of higher value of PCR of source T12 in Figure 3.3 in case of EPRCA algorithm. Here throughput curve of source T12 is much above the throughput curves of other sources of the same group. We can observe corresponding results for EPRCAM algorithm in Figure 3.3b. As expected all curves are almost coinciding with each other indicating that there is no unfairness due to higher PCR value of source T12.

3.7.3 Better and Fair Throughput

Table 3.1 provides a summary of throughput and fairness results of all sources for EPRCA and EPRCAM algorithms for link distances of 120Km and PCR=30Mbps.

Comparing columns of unfairness results we can see that in case of EPRCA, unfairness in bandwidth utilization is around 20% in favor of cross traffic. On the other hand, for EPRCAM, it is only around 10% in favor of cross traffic. Now, looking at the rows with labels "L8", "L12" and "L16", we observe that the maximum link utilization for EPRCA is 91.4%, however, it is 97.4% for EPRCAM algorithm, resulting in a 6% gain in throughput.

To further clarify the impact of the problems of EPRCA discussed in section 3.5, another simulation was run with peak cell rate of each source quite large that is 60Mbps so that sources could increase their rates as much as allowed in the no congestion state. The throughput and fairness results for both EPRCA and EPRCAM for this scenario are summarized in Table 3.2.

We can see from this table that increasing PCR values of sources from 30Mbps to 60Mbps does not affect performance of EPRCAM, however, in case of EPRCA unfairness towards end-to-end traffic increases. For example, on average, cross traffic utilizes 20% more bandwidth than the end-to-end traffic when source PCR values are 30Mbps. On the other hand, when source PCR values are increased to 60Mbps, the bandwidth utilization by the cross traffic becomes around 40% more than the end-to-end traffic. The reason behind this large unfairness in favor of cross traffic in case of EPRCA is clear from Figures 3.4 & 3.5. Each one of these figures shows source rates for T0 (end-to-end traffic) and T4 (cross traffic). We can observe from Figures 3.4a & 3.5a that in case of EPRCA, the cross traffic keeps on increasing its rates up to PCR values in no congestion state whereas end-to-end traffic seldom gets a chance to reach its PCR value. On the other hand we see that in case of EPRCAM algorithm, all source rates are maintained close to the operating point of the network even in no congestion state (Figures 3.4b & 3.5b).

3.7.4 Impact of RIFS

To clarify impact of the newly introduced RIFS parameter in EPRCAM, simulations were repeated for four different values of RIFS parameter; that is 1/8, 1/16, 1/32 and 1/64. The throughput and unfairness results (of EPRCAM) are given in Table 3.3.

As expected (refer section 3.6.5) we see that as the value of RIFS is decreased from 1/8 to 1/64, the unfairness in favor of cross traffic decreases. For example, looking at the table, the maximum unfairness in favor of cross traffic for RIFS=1/8 is 14%, however, it is 8% for RIFS=1/16. At the same time, we see that total link utilization has degraded from 97.8% to 96.8% (1% degradation). Though, the optimal value of RIFS may depend on network topology, we recommend a value of 1/16 for RIFS which seems to produce near optimal results.

3.8 Summary & Conclusion

In this chapter, I identified problems with the EPRCA algorithm which cause poor throughput and fairness performance and proposed a Modified EPRCA (EPRCAM). The EPRCAM was evaluated against EPRCA and it showed much better throughput and fairness performance. The impact of the newly introduced parameter RIFS (Rate Increase Factor for Switch) has minimal impact on both throughput as well as fairness performance of the network. I recommend a value of 1/16 of RIFS for near optimal performance.

I have extensively evaluated the performance of EPRCAM against congestion avoidance algorithms like ERICA and have submitted the manuscript to an international conference for presentation.

I conclude as follows.

- EPRCA causes underutilization and unfairness due to a combination of no control exerted by the switch in no congestion state and high reductions done in the high congestion state.
- This can be avoided by the Modified EPRCA (EPRCAM) which lets a switch control rates of sources in congestion states as well as in no congestion state resulting in complete synchronization of source rates.

Table 3.1: Throughput and Unfairness Results of EPRCA & EPRCAM (PCR=30Mbps, Link=120Km)

	Throughput		*Unfairness*	
VC	*EPRCA*	*EPRCAM*	*EPRCA*	*EPRCAM*
T0	0.124	0.134	03	01
T1	0.120	0.133	00	00
T2	0.120	0.133	00	00
T3	0.121	0.134	01	01
T4	0.144	0.144	20	08
T5	0.144	0.143	20	08
T6	0.142	0.145	18	09
T7	0.140	0.147	11	11
T8	0.143	0.146	19	10
T9	0.143	0.147	19	11
T10	0.136	0.148	13	11
T11	0.145	0.147	21	11
T12	0.140	0.146	17	10
L8	0.914	0.964		
L12	0.910	0.974		
L16	0.905	0.974		

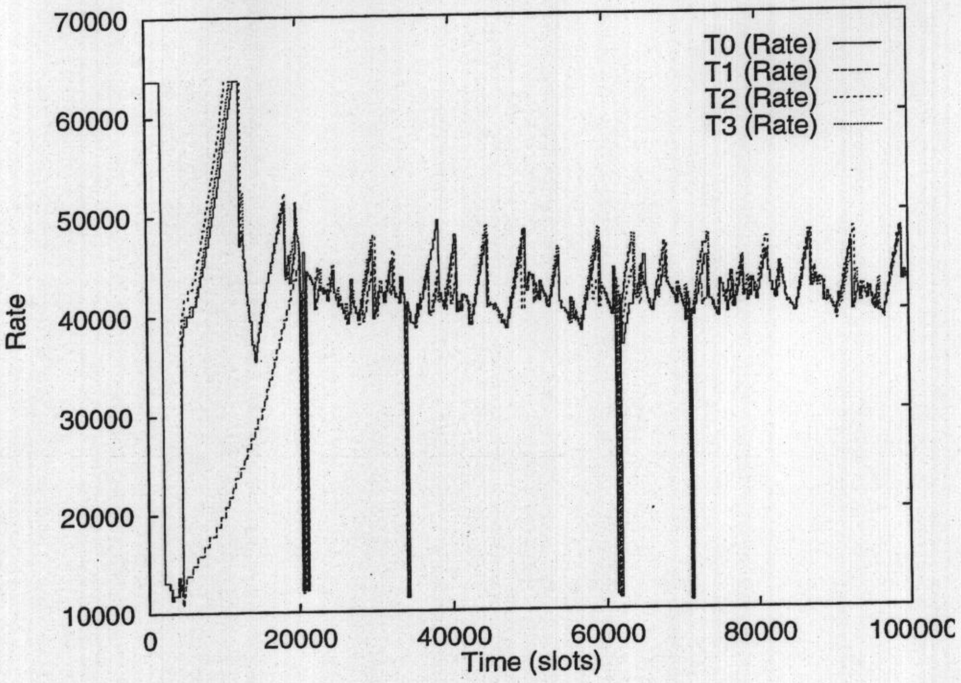
Table 3.2: Throughput and Unfairness Results of EPRCA & EPRCAM (PCR=60Mbps, Link=120Km)

	Throughput		*Unfairness*	
VC	*EPRCA*	*EPRCAM*	*EPRCA*	*EPRCAM*
T0	0.111	0.134	01	02
T1	0.121	0.135	10	02
T2	0.121	0.132	10	00
T3	0.110	0.133	00	01
T4	0.140	0.146	27	11
T5	0.148	0.145	35	10
T6	0.137	0.144	25	09
T7	0.150	0.144	36	09
T8	0.144	0.146	31	11
T9	0.148	0.147	35	11
T10	0.145	0.147	32	11
T11	0.157	0.146	43	11
T12	0.152	0.148	38	12
L8	0.888	0.968		
L12	0.904	0.971		
L16	0.917	0.975		

Table 3.3: Throughput and Unfairness Results of EPRCAM showing impact of RIFS Parameter (PCR=60Mbps, Link=120Km)

	Throughput				*Unfairness*			
VC	*1/8*	*1/16*	*1/32*	*1/64*	*1/8*	*1/16*	*1/32*	*1/64*
T0	.132	.136	.134	.132	00	2	1	0
T1	.134	.136	.134	.133	02	2	1	1
T2	.134	.133	.133	.134	02	0	0	2
T3	.134	.136	.136	.134	02	2	2	2
T4	.144	.141	.137	.134	09	6	3	2
T5	.146	.141	.136	.135	11	6	2	2
T6	.143	.139	.139	.134	08	5	5	2
T7	.147	.142	.138	.136	11	7	4	3
T8	.146	.142	.139	.135	11	7	5	2
T9	.145	.141	.140	.137	10	6	5	4
T10	.150	.141	.141	.137	14	6	6	4
T11	.146	.143	.139	.137	11	8	5	4
T12	.148	.143	.142	.136	12	8	7	3
L8	.966	.963	.950	.936				
L12	.973	.966	.955	.942				
L16	.978	.968	.960	.944				

a. EPRCA(Rates)



b. EPRCAM(Rates)

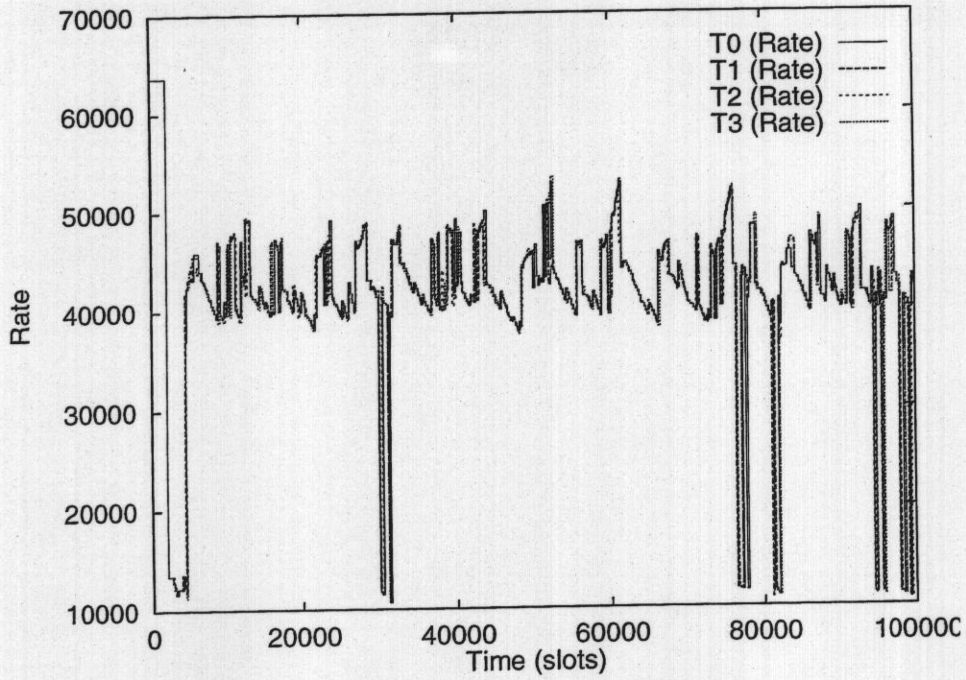
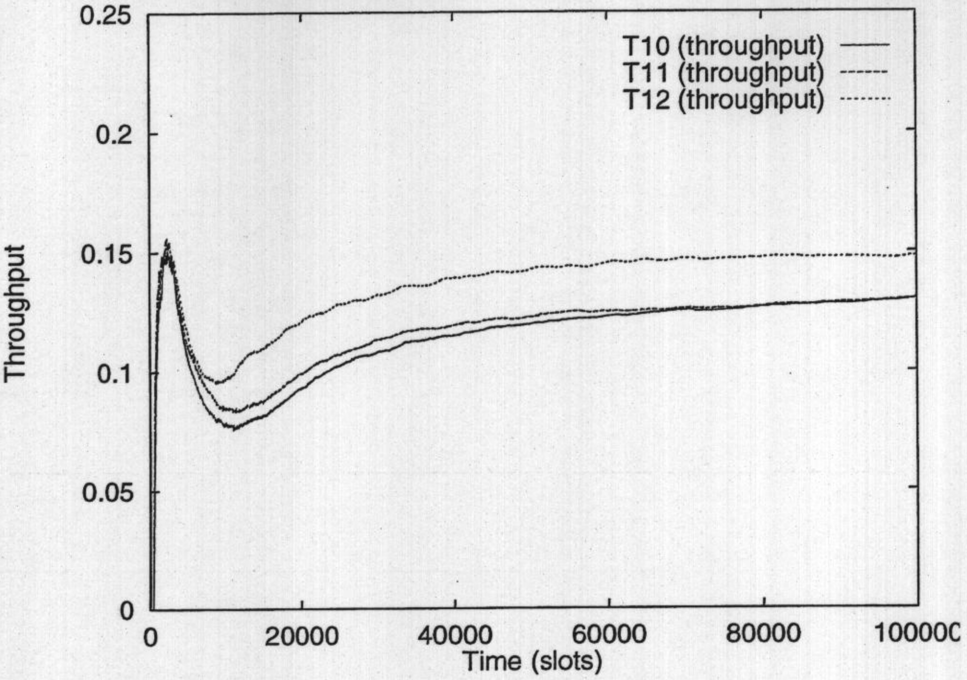


Figure 3.2: Comparison of EPRCA and EPRCAM in terms of Rate Control Performance (Each Link = 120 km, PCR=30Mbps)

a. EPRCA(Throughput), PCR of T12=60Mbps



b. EPRCAM(Throughput), PCR of T12=60Mbps

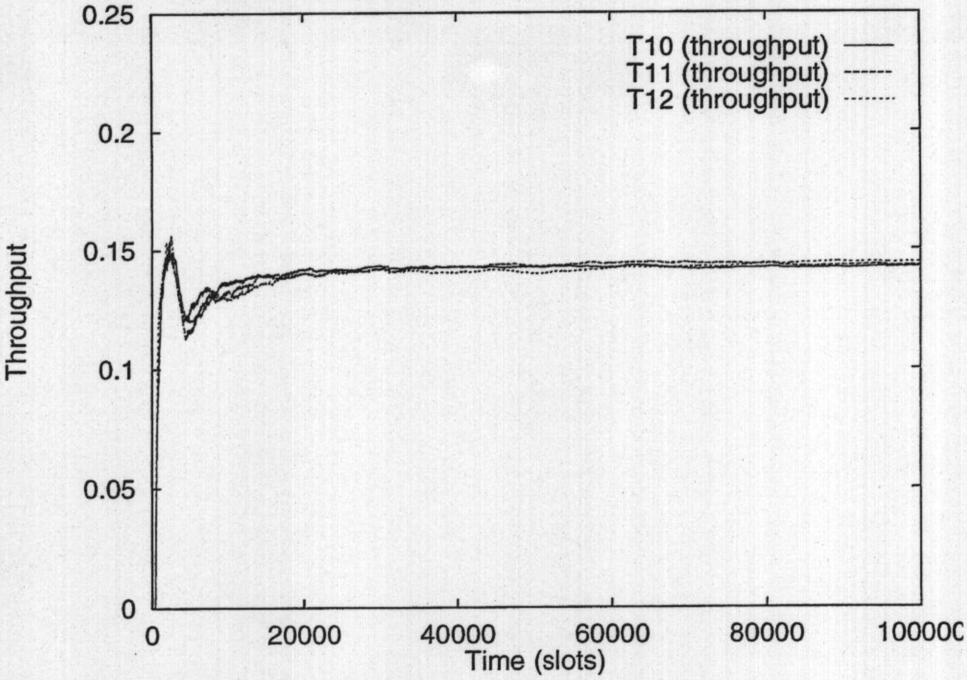
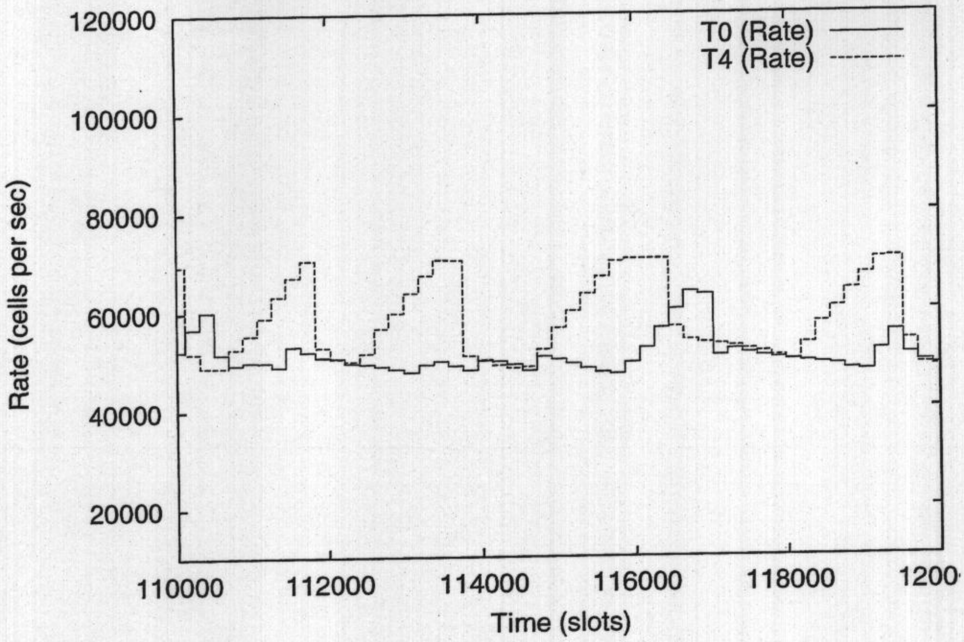


Figure 3.3: Comparison of EPRCA and EPRCAM Throughput Performance (Each Link = 120 km, PCR=30Mbps)

a. EPRCA, Pcr=30Mbps



b. EPRCAM, Pcr=30Mbps

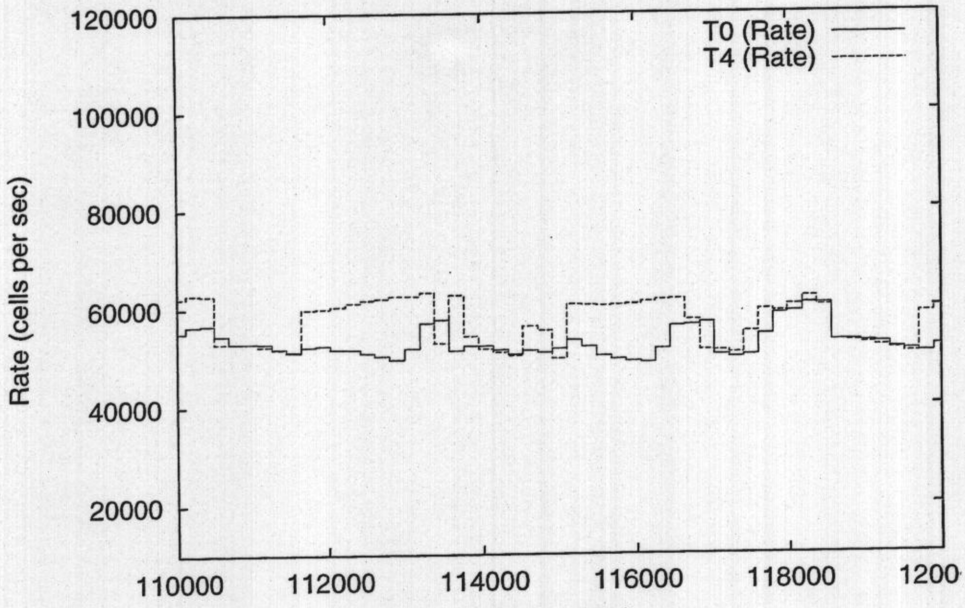
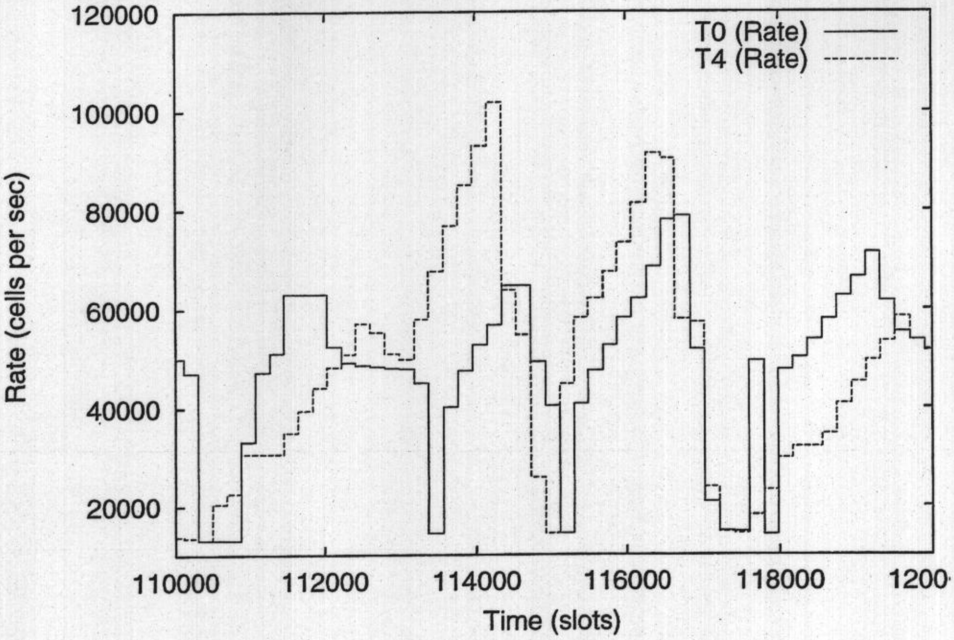


Figure 3.4: Comparison of EPRCA & EPRCAM in terms of impact of PCR (Link=120km, PCR=30Mbps)

a. EPRCA, Pcr=60Mbps



b. EPRCAM, Pcr=60Mbps

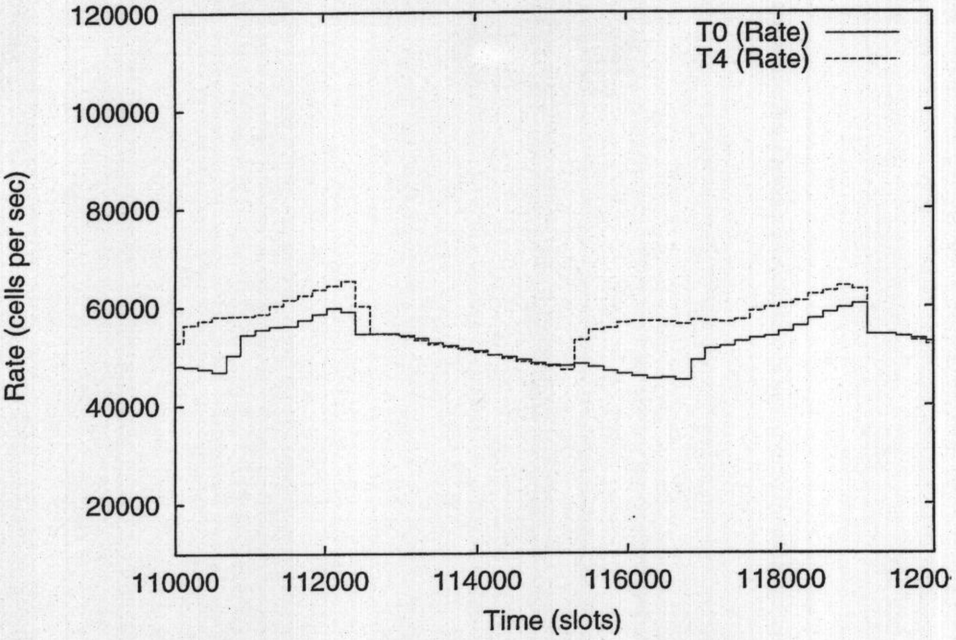


Figure 3.5: Comparison of EPRCA & EPRCAM in terms of impact of PCR (Link=120km, PCR=60Mbps)

Chapter 4

Improved Fairness Performance

4.1 Introduction

The Available Bit Rate (ABR) and Unspecified Bit Rate (UBR) services of Asynchronous Transfer Mode (ATM) networks are designed to divide fairly and efficiently the bandwidth not used by other service classes. Both of these services are designed for data traffic, however, ABR provides better and fair distribution of bandwidth among contending sources and ensures minimum cell loss because of its traffic control.

The traffic control mechanism basically consists of a set of rules (congestion control/avoidance algorithm) used by switches to calculate rate of transmission for sources passing through them to ensure fair and efficient utilization of available bandwidth.

Many explicit rate notification (congestion control/avoidance) algorithms have been proposed. The most famous and extensively discussed at present are EPRCA (Enhanced Proportional Rate Control Algorithm) and ERICA (Explicit Rate Indication for Congestion Avoidance).

I identified some deficiencies with EPRCA algorithm and suggested that it needs some changes to improve performance. I called the modified algorithm as EPRCAM (Modified Enhanced Proportional Rate Control Algorithm). I showed that the modified algorithm achieves better and fair available bandwidth utilization [44].

In this chapter, I further our study to ERICA and will establish that the tuning problem of **measurement interval** and **queue control factor** of ERICA may cause high variance in measurements of load factor and available ABR capacity which results in unfairness. We will show that unlike claims of the proponents of ERICA, it does not achieve Max-Min fairness. I have discovered unfairness in favor of cross traffic in ERICA. I show that, due to continuous monitoring of the network state by EPRCAM in the form of MACR (Mean of Allowed Cell Rates), an exponential average of current cell rates of transmitting sources, it produces less unfairness in favor of cross traffic. We will also show that EPRCAM is much simpler to implement than ERICA.

The remainder of this chapter is organized as follows. In section 4.2, I will briefly describe EPRCA and its main deficiencies. In section 4.3, I will describe EPRCAM algorithm. The tuning problem of *interval problem* and *queue control factor* of ERICA which cause high variance in measurement of essential parameters and thus results in unfairness in favor of cross traffic will be described in section 4.4. EPRCAM and ERICA will be evaluated and compared in section 5.6. Finally, I will summarize in section 4.6.

4.2 Enhanced Proportional Rate Control Algorithm (EPRCA)

In EPRCA, a switch maintains a moving exponential average of CCR (Current Cell Rate) values taken from the Forward Resource Management (FRM) cells of all the virtual connections (VC) passing through it. This average is called Mean of Allowed Cell Rates (MACR). When Backward RM (BRM) cells cross a switch and if the switch is congested, it calculates explicit rate for the VC of the passing BRM using MACR value.

The switch maintains three states namely *No_Congestion*, *Low_Congestion* and *High_Congestion* based on two thresholds that is, a low threshold (*Low_Threshold*) and a high threshold (*High_Threshold*). The states of a switch are decided depending on the length of queue in the switch (Switch_Queue) as per following expressions.

```
IF (Switch_Queue>=High_Threshold) THEN
     High_Congestion=TRUE
IF (Switch_Queue>=Low_Threshold AND Switch_Queue<High_Threshold) THEN
     Low_Congestion=TRUE
```

The explicit rate (Explicit_Rate) for a source is calculated according to the following algorithm.

```
IF (High_Congestion) AND (Explicit_Rate>=MACR*MRF) THEN
        Explicit_Rate = Min(Explicit_Rate,MACR*MRF)
ELSE IF (Congestion) AND (Explicit_Rate>=MACR*DPF)
        Explicit_Rate = Min(Explicit_Rate,MACR*ERF)
END (*IF*)
```

The typical values for the ERF (Explicit Reduction Factor) and MRF (Major Reduction Factor) are 15/16 and 1/4 respectively. The above algorithm means that the switch provides no response in the No_Congestion state while in Low_Congestion and High_Congestion states a switch notifies its sources to reduce their rates by fractions of ERF and MRF respectively.

4.2.1 No-Control-In-No-Congestion-State Problem

In an stable and a synchronized source rates, no Explicit_Rate indication in No_Congestion state seems to be logical but in the real world, there are sources with different propagation delays resulting in different source rates. Moreover, due to major reductions in source rates in High_Congestion state, there is a possibility of highly diverse source rates when a switch goes into the No_Congestion state which often follows a High_Congestion state. The whole phenomenon produces unfairness in bandwidth allocation.

Another problem caused by the No-Control-In-No-Congestion-State approach is that in No_Congestion state, sources increase their rates as a fraction of their peak cell rate (PCR). This in turn causes unfairness in bandwidth utilization if different sources have different PCR values.

Note that the problems caused by the approach of No-Control-In-No-Congestion-State are described in detail in [44].

4.3 Modified Enhanced Proportional Rate Control Algorithm (EPRCAM)

To solve problems of EPRCA caused by the No-Control-In-No-Congestion-State described in section 4.2, I proposed in [44] to modify EPRCA so that a switch should notify Explicit_Rate to its sources in all states including No_Congestion state. I call this algorithm Modified EPRCA (EPRCAM). The complete modified algorithm is given below:

```
IF (High_Congestion) AND (Explicit_Rate>=MACR*MRF) THEN
        Explicit_Rate = Min(Explicit_Rate,MACR*MRF)
ELSE IF (Low_Congestion) AND (Explicit_Rate>=MACR*DPF)
        Explicit_Rate = Min(Explicit_Rate,MACR*ERF)
ELSE IF (Explicit_Rate<MACR*DPF)
        Explicit_Rate = Min(Explicit_Rate,MACR*(1+RIFS))
END (*IF*)
```

where, RIFS = Rate Increase Factor for Switch.

The modified algorithm effectively says that in No_Congestion state, a switch should notify the sources passing through it to bring their rates to MACR*(1+RIFS). The behavior in the other two states namely, High_Congestion state and Low_Congestion state is the same as EPRCA.

The general behavior of EPRCAM is such that in stable environment, switch queues are made to fluctuate near the Low_Threshold by increasing and decreasing their rates. However, due to some statistical fluctuation or introduction of new sources, a switch may go into High_Congestion state. Here due to notification of major reductions and reasons specified in [44], the source rates may be significantly different even in case of the same distance sources. In No_Congestion state which usually follows a High_Congestion state, the high rate sources as well as low rate sources are all notified to synchronize their rates to MACR*(1+RIFS). Remember that after major reductions in source rates, the switch should already have adjusted (reduced) its MACR to the value it can support. This eliminates unfairness which may be caused due to diversified source rates.

4.3.1 Fairness Issue

I established in [44] through arguments and simulation results that due to its explicit rate notification in all three states, EPRCAM eliminated unfairness caused due to No-Control-In-No-Congestion-State problem of EPRCA.

I now further my study to include ERICA and present in the subsequent sections the problems of ERICA which cause unfairness towards end-to-end traffic in bandwidth allocation and compare the fairness performance of ERICA with EPRCAM.

4.4 Explicit Rate Indication for Congestion Avoidance (ERICA)

ERICA [53, 54, 55, 56] is a congestion avoidance algorithm for ABR traffic which computes explicit rate to be indicated to sources in RM cells. ERICA consists of a few options besides the following basic steps.

An ERICA switch periodically monitors load on each link and determines a load factor, z, the available capacity, and the number of currently active VCs. The load factor is calculated as the ratio of the measured input rate at the port to the target capacity of the output link:

$$z = (\text{ABR Input Rate})/(\text{ABR Capacity}),$$

where: ABR Capacity = Target Utilization x Link Bandwidth - VBR Usage - CBR Usage

The input rate is measured over an interval called the *switch measurement interval.* In basic ERICA algorithm, the target utilization is set to a fraction (close to, but less than 100%) of the available capacity. However, in ERICA+, the target utilization is dynamically changed depending on the filling of switch queues. It is the ERICA+ that I am targeting in this chapter so that I shall denote ERICA+ as ERICA for simplicity.

The load factor, z, is an indicator of the congestion level of the link. The optimal operating point is at a load factor value of one.

The fair share of each VC, *FairShare*, is computed as follows:

FairShare=(ABR Capacity)/(Number of Active Connections).

Each connection can send at a rate up to *FairShare.* If a connection does not use all of its *FairShare*, then the switch fairly allocates the remaining capacity to other connections using a quantity called *VCShare* computed as follows:

VCShare=CCR/z.

The explicit rate (Explicit_Rate) is calculated as:

Explicit_Rate=Max(FairShare,VCShare).

If the *VCShare* value is greater than *FairShare* value, the source is allowed to send at *VCShare.*

4.4.1 Fairness Problem Due to High Variance

Unlike initial claims of better and fair throughput by the proponents of ERICA, it has now been admitted to have tuning of *measurement interval problem* which may cause unfairness. I have found that a combination of the measurement interval problem and queue control factor leads to high variance in the measurement of important parameters like load factor and available capacity. I will describe these problems related with ERICA in next subsections. In section 4.5, I will quantitatively evaluate ERICA and show that the severity of fairness problem may be more than anticipated and that EPRCAM may outperform ERICA in terms of fairness in bandwidth utilization and implementation complexity.

Impact of Measurement Interval

ERICA measures quantities like load factor, z, the available capacity, and the number of currently active VCs over consecutive intervals and uses these measured quantities in each interval to calculate feedback in the next interval. Performance of ERICA largely depends on the accuracy of the measurement of these quantities which in turn depends on the size of the *measurement interval.* In [47], the author admits that; *"Longer intervals produce better averages, but slow down rate of feedback. Shorter intervals may result in more variation in measurements, and may consistently underestimate or overestimate the measured quantities"*. The estimation of the size of interval is an extremely complex task. For example, the size of the interval depends on the link speed and the number of VCs. The link speed may be known in advance but the number of VCs may not be known. Even an estimate of VCs may be difficult to make because the active number of VCs may be different during morning, afternoon and night time in a network.

Though, a new algorithm has been proposed in [47] for estimation of number of active VCs by dividing VCs into *underloading* and *overloading* on the basis of their *activity level,* it is not completely insensitive to *measurement interval.* Moreover, the new algorithm requires several iterations for correct estimation.

Impact of Queue Control Factor

The author says in [47], that "**assuming that the measurements do not exhibit high variance** *and the measurement interval is long enough to estimate the number of VCs, the load factor and the available capacity, the original ERICA converges to efficient operation in all cases"*.

I consider this assumption as a very important one. Note that in the original ERICA, 5-10% of available ABR capacity is used for draining of queues. However, in ERICA+ algorithm, the share for draining of queues dynamically varies with the size of switch queues depending on a factor called **queue control factor**, which in turn makes the available ABR capacity vary dynamically. Now, if the queue control factor is not tuned properly, this may result in high variance in the available ABR capacity which may result in unsatisfactory fairness and throughput results.

Moreover, according to Jain [47], *VCShare* aims at bringing the system to an efficient point which may not necessarily be fair. Note that the high variance caused by a possible improper tuning of queue control factor will result in frequent changes in the load factor and thus in the *VCShare.* Since, the end-to-end traffic passes through all of the switches in a network and faces longer delays, it may not be able to use its *VCShare* as quickly as the cross-traffic. Thus *VCShare* makes cross traffic too greedy and produces unfairness in their favor.

In the section 4.5 I will examine the impact of the *VCShare* and high variance due to queue control factor on the fairness performance of ERICA using simulations of the parking lot network model as shown in Figure 4.1. I show that high variance caused in estimation of *VCShare* due to a combination of *measurement interval* and *queue control factor* may lead to large amount of unfairness in favor of cross traffic.

4.5 Comparison between EPRCA, EPRCAM and ERICA

4.5.1 Implementation Complexity

As already stated in section 5.2, the difference of EPRCAM from EPRCA is the explicit notification given by an EPRCAM switch in the No_Congestion state as well. Since, the basic operations required by both of the algorithms are the same, there is no difference in the implementation complexity of EPRCA and EPRCAM.

Let us now compare EPRCAM with ERICA in terms of processing power and other requirements.

The basic operations required by EPRCAM are:

- To maintain a moving average of CCR values taken from FRM cells.
- To determine state of switch by monitoring two thresholds on the switch queue.
- To calculate explicit rate for a VC using a simple multiplication.

The basic operations required by ERICA are:

- Measurement interval: requires maintenance of a timer.
- Number of active VCs: requires a table, counters and exponential averaging.
- Load factor: requires a division and exponential averaging.
- ABR capacity: requires a division.
- Fair share: requires a division.
- VC share: requires a division.
- Queue control factor: requires monitoring of queues.
- CCR: a table is to be maintained.

Obviously the number of operations is much greater for ERICA than EPRCAM. It is well established that at the current state of technology the floating point divisions use more number of basic processor operations than multiplications. Obviously ERICA uses a number of divisions, whereas EPRCAM uses only a few multiplications.

Moreover, two per VC tables have to be maintained by ERICA, one each for active VCs and CCR values. These tables are essential because measurements are done at the end of interval and thus status of each VC has to be recorded using these tables. Note that whether the CCR is calculated by the switch or taken from RM cells does not make any difference on the necessity of tables. The size of these tables increase with the number of VCs. On the other hand, EPRCAM does not require a CCR table because it uses CCR values for the computation of MACR only and the CCR is not needed for later use.

In the following subsection I will describe the network model on which I performed simulations in order to compare performance of EPRCA, EPRCAM & ERICA.

4.5.2 Network Simulation Model

Description of Model & Parameter Setting

As depicted in Figure 4.1, our network simulation model is a multihop wide area network with four switches (4x4, output buffer type), thirteen sources and three destinations. The sources are divided into four groups. The dashed lines show the traffic flow. The purpose of choosing such a complex model is to make it as realistic as possible.

Group-A: Source T0 to Source T3 (end-to-end traffic)
Group-B: Source T4 to Source T6 (two-hop traffic) (cross-traffic)
Group-C: Source T7 to Source T9 (two-hop traffic) (cross-traffic)
Group-D: Source T10 to Source T12 (one-hop traffic) (cross-traffic)

The values of parameters used are given below.

- Low_Threshold: 20 cells, High_Threshold: 100 cells
- link capacity = 155.52 Mbps
- Peak Cell Rate (PCR) = 30Mbps, 60Mbps
- Initial Cell Rate (ICR) = PCR/2
- Each Link = 60 Km
- Simulation Time = 5 secs

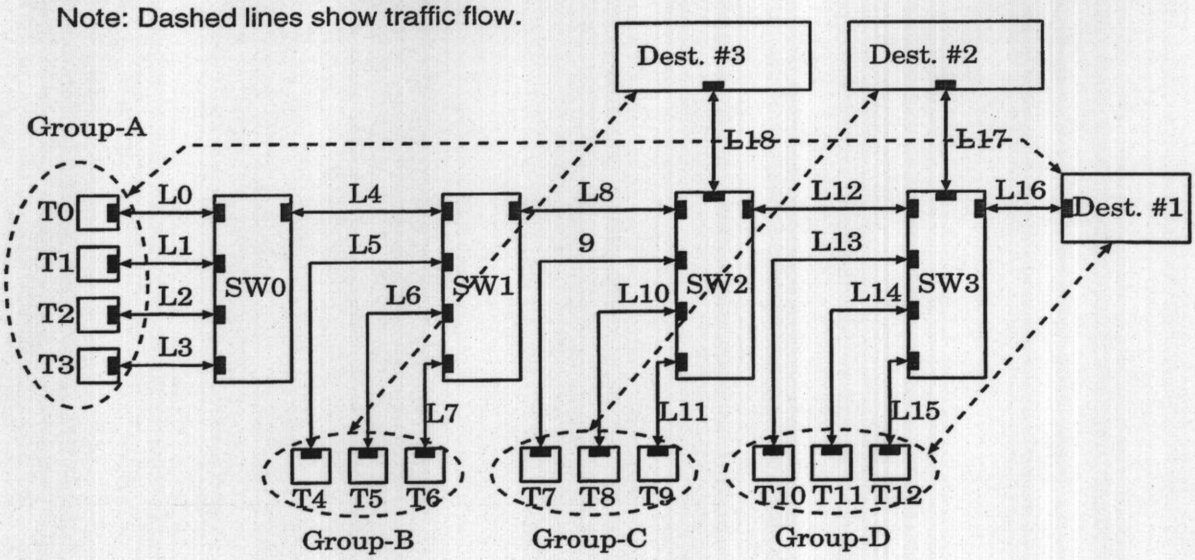

Note: Dashed lines show traffic flow.

Figure 4.1: Network Simulation Model

Notes:

- Gready source model: sources always have data to be sent.
- Looking at the traffic pattern, *max-min throughput* [48] of all sources is 0.143.
- Looking at the traffic pattern, seven VCs pass through each of L8, L12 and L16 links.
- Other parameters are taken as recommended by the ATM Forum in [48].

Queue Control Factor in ERICA

According to the specification of ERICA [56], the queue control factor can be user dependent so we used three simple methods to calculate queue control factor and to see their impact on the performance of ERICA.

Method1: Queue control factor is designed in such a way that a small portion (5%) of available ABR capacity is used for draining of queues when switch queues are between Low_Threshold ad High_Threshold. On the other hand a very larger ABR Capacity (75%) is allocated to quickly drain switch queues when switch queues are above High_Threshold. This can be summarised by the following expressions.

```
IF(Queue<=Low_Threshold) THEN
```

```
    ABR_Capacity=100% of ABR_Capacity
IF(Queue>Low_Threshold AND Queue<High_Threshold) THEN
    ABR_Capacity= 95% of ABR_Capacity
IF(Queue>High_Threshold) THEN
    ABR_Capacity= 25% of ABR_Capacity
```

Method2: In this method we reserve a comparatively larger portion (10%) of capacity when queue are between Low_Threshold and High_Threshold while only 50% of ABR capacity is reserved when queue are above High_Threshold. The assumption behind this method is that 5% capcity may not be sufficient in WAN environments to quickly drain queues.

```
IF(Queue<=Low_Threshold) THEN
    ABR_Capacity=100% of ABR_Capacity
IF(Queue>Low_Threshold AND Queue<High_Threshold) THEN
    ABR_Capacity= 90% of ABR_Capacity
IF(Queue>High_Threshold) THEN
    ABR_Capacity= 50% of ABR_Capacity
```

Method3: Like *Method2*, 10% of available ABR capacity is used for queue drainage when switch queues are above Low_Threshold. However, High_Threshold is not taken into account at all.

```
IF(Queue<=Low_Threshold) THEN ABR_Capacity=100% of ABR_Capacity
IF(Queue>Low_Threshold)  THEN ABR_Capacity= 90% of ABR_Capacity
```

4.5.3 Performance Comparison

In this section I will do a performance evaluation of EPRCA, EPRCAM & ERICA. Though I will provide results of EPRCA, our discussion will mainly remain focused on EPRCAM & ERICA. For comparison and discussion between EPRCA & EPRCAM, please refer [44].

4.5.4 Fairness

Definition: Since, a strict definition of unfairness is not available, I give my own definition as follows.

$$F_n(\%) = (T_n - T_i)/T_i * 100$$

such that $T_i \leq T_j$ For $j = 0, 1, ..., N$
where, F_n = Unfairness in % for source n,
and, T_n = Normalized actual throughput of source n.

In the above expression, $F_n(\%)$ gives the difference between the throughput of a source and that of the source achieving minimum throughput. The condition in the "where" clause states that the fairness comparison can only be done between sources which expect the same throughput (according to max-min criteria in [48] page 90).

4.5.5 Impact of Measurement Interval

To see the impact of *measurement interval* as described in section 4.4.1 on the fairness performance of ERICA switch, simulations were run for measurement intervals of 20, 40, 60, 80, 100, 200 and 1000 μ sec. The throughput and fairness results obtained are summarized in Table 4.1.

During simulation, it was observed that for a measurement interval of 20 μ sec, the estimation of the number of active VCs through the links L8, L12 and L16 was ranging from 1 to 6 VCs. Whereas, when the measurement interval was doubled to 40 μ sec, the estimation improved and was from 4 to 7 VCs. Note that the actual number of VCs passing through L8, L12 and L16 is 7. Further increase in the measurement interval, improved estimation of the active number of VCs and at around 80 to 100 μ sec, correct number of VCs was being estimated.

In ideal conditions, measurement interval can be calculated as,

```
Interval = Cell Time * Number of Setup VCs
         = 2.72 (micro-sec) * 7 = 19.04 micro-sec
```

However, due to sources with different propagation delays and different transmission rates, the size of interval required to correctly estimate active number of VCs may be large. In our simulation it seems to be around 100 μ sec.

Looking at the Table 4.1, I can observe that the unfairness exists in favor of cross traffic for all measurement intervals. This is due to variations in estimations (number of active VCs, ABR capacity, Load Factor). In case of variation, cross traffic is quick to respond, and in particular becomes very greedy due to the VCShare. Note that the end-to-end traffic due to its long delay and passing through all hops may not be able to use its *VCShare* so quickly. However, as the estimation of number of active VCs improves, so does the unfairness. For example, column 2 of Table 4.1 shows that Group-D sources get around

40% more throughput than the Group-A sources when the measurement interval is 20 μ sec. On the other hand it is only around 20% more when the measurement interval is 100 μ sec.

Hence, fairness performance of ERICA depends on the proper tuning of the measurement interval. On the other hand EPRCAM does not depend on any time interval and monitors the network state by maintaining an exponential average whose computation is quite simple and is done in real time (section 5.2).

4.5.6 Impact of Queue Control Factor

To eliminate variance in the estimations of number of active VCs, ABR capacity and Load Factor, the measurement interval was made sufficiently larger (1000 μ sec) than the optimal estimate of 100 μ sec as discovered in section 4.5.5. Simulation was run for two different values of source PCRs that is; (i) Each source PCR = 30Mbps, (ii) Each source PCR = 60Mbps. The results of all of the three algorithms are given in Table 4.2.

I can see from columns 3 & 6 of EPRCAM that there is no impact of PCR values of sources on throughput or fairness performance of sources. However, there is a 10% unfairness towards end-to-end traffic in both cases due to long delays and passing through all hops. Note that since EPRCAM does not use CCR value of a source when deciding explicit rate, source rates do not observe large diversity in rate allocations.

On the other hand, columns 4 & 7 of ERICA show that the unfairness in favor of cross traffic increased from around 15% to around 80% when source PCRs were increased from 30Mbps to 60Mbps. On careful observation of different parameters of ERICA during simulation we found the following explanation. *Method1* (section 4.5.2) was used to compute queue control factor for results given in Table 4.2. In *Method1* only 5% capacity is allocated to drain queue when queues are between Low_Threshold and High_Threshold which proved to be insufficient in our model to prevent queues to fill beyond High_Threshold. When queues are above High_Threshold, 75% of available capacity is used to drain queue. This typical approach of *Method1* makes available capacity vary heavily. Since, other parameters like load factor and VCShare are dependent on available capacity, they too observe large variance. As already described in the previous section, variance in VCShare results in unfairness in favor of cross traffic.

Above stated phenomenon can be seen in Figures 4.2 & 4.3. They show source rates from time slot 110000 to time slot 120000 only. The portion was selected randomly. Figures 4.2a, b and c give source rates of T0 (end-to-end traffic) and T4 (cross-traffic) for EPRCA, EPRCAM and ERICA/ERICA+ respectively when PCR values of sources are 30Mbps or

74191 cps (cells per sec). Corresponding results for PCR values of 60Mbps or 148383 cps are given in Figures 4.3a, b and c.

We can clearly see that for all algorithms most of time the curves for cross-traffic (T4) float above end-to-end traffic showing larger allocations for cross-traffic. Looking at Figures 4.2b and 4.3b for EPRCAM we can see that there is almost no difference of allocation when the source PCR is changed from 74191 cps to 148383 cps. The allocations for T0 and T4 remain around 60000 cps. Let us now examine the rate behavior for ERICA in Figures 4.2c & 4.3c. In these figures, we can see that the curve of T4 goes up to the PCR values that is 74191 cps and 148383 cps respectively, showing that *VCShare* has been larger than even PCR value while the curve for T0 remains around or below *FairShare*. This clearly shows the impact of high variance due to queue control factor on *VCShare* and in turn on fairness performance of the network.

To further investigate our argument, we ran a simulation for ERICA for all three methods (*Method1, Method2 & Method3*) without changing other parameters. In Method2 & Method3, 10% of available capacity is allocated to drain queues which queues from filling beyond High_Threshold and thus substantially reduces variance of other parameters. The results are summarized in Table 4.3. Looking at columns 7, 8 & 9, we can see that unfairness has been substantially reduced from around 80% to just around 15% in favor of cross traffic.

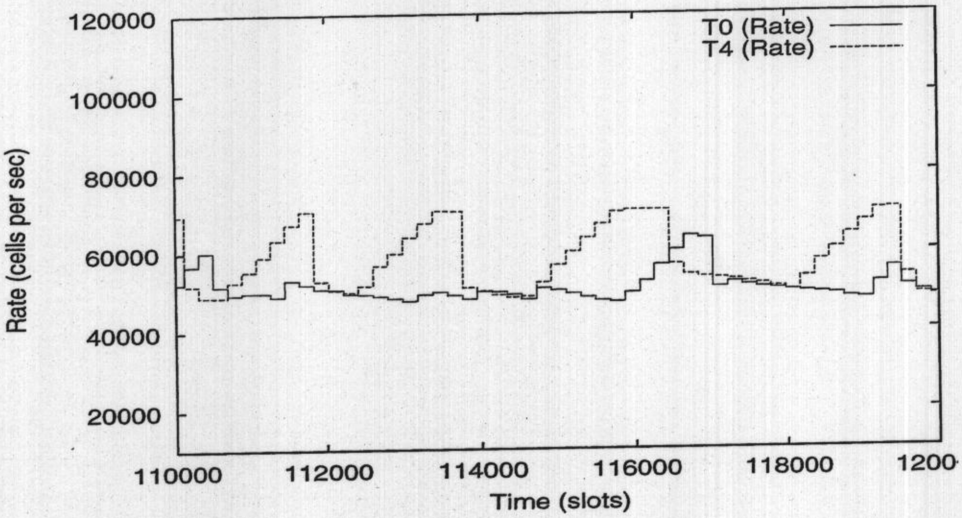
4.6 Summary

In this chapter I first identified problems with EPRCA algorithm which lead to unfairness in available bandwidth utilization by the contending ABR sources in favor of cross traffic. I described Modified EPRCA (EPRCAM) algorithm which eliminates these problems and leads to substantial gain in throughput and fairness performance.

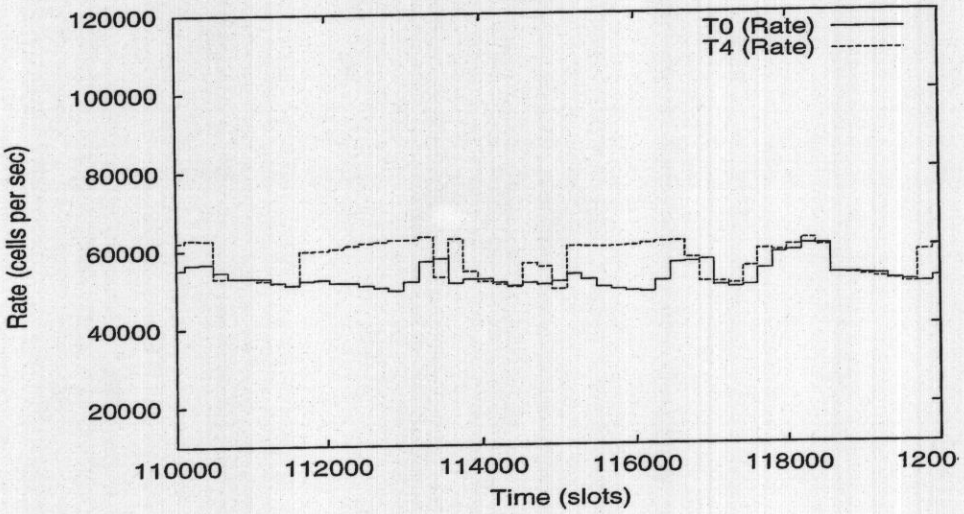
I then examined the ERICA algorithm which is one of the most extensively discussed ABR traffic control algorithm at present. We established through arguments and simulation results that ERICA algorithm may cause substantial unfairness in favor of cross traffic. I suggested that the following two parameters are primary cause of resulting in high variance in the estimation of VC share and thus producing unfairness in favor of cross traffic.

- Difficulty in tuning the length of measurement interval.
- Queue control factor.

I compared and contrasted ERICA and EPRCAM algorithms and quantitatively showed that EPRCAM gives better fairness and throughput performance than ERICA mainly due *a. EPRCA*



b. EPRCAM



c. ERICA

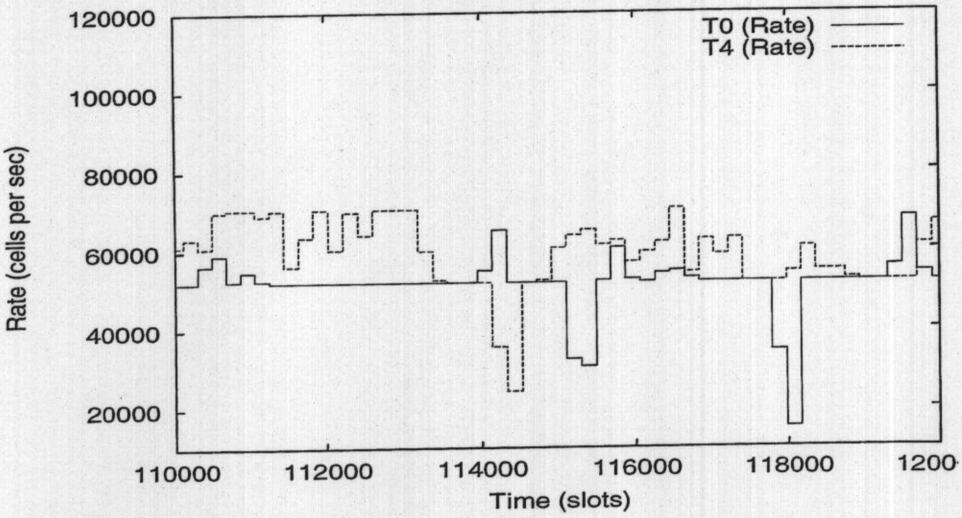
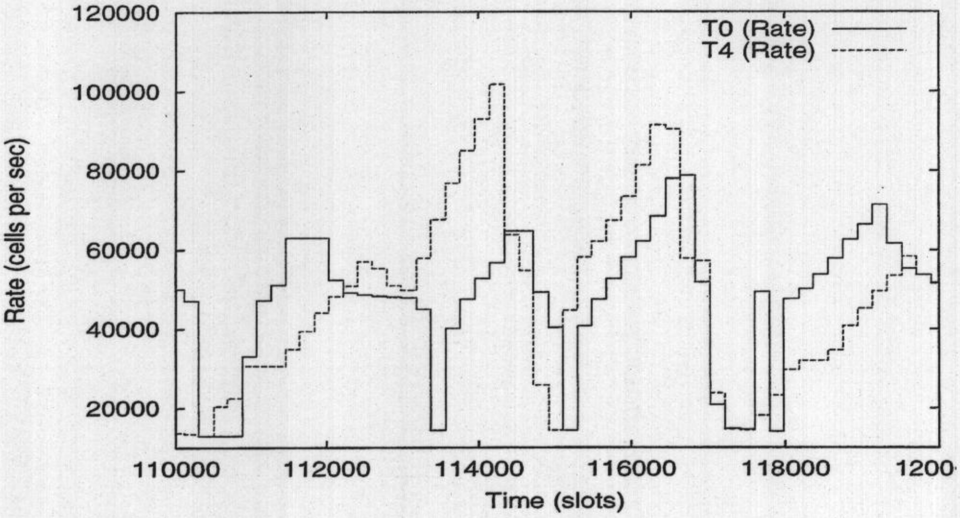
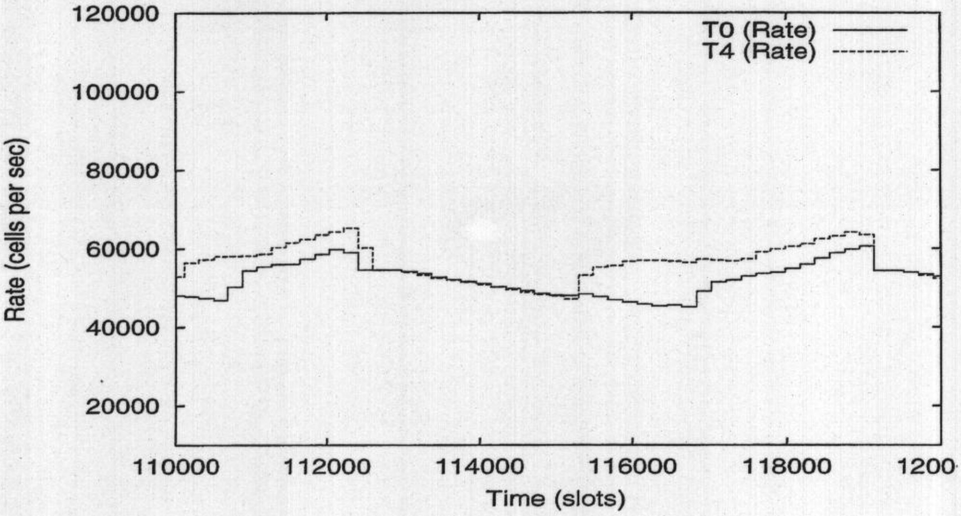


Figure 4.2: Results of ERICA showing unfairness in favor of cross traffic due to high variance in VCShare and Load Factor caused by the Queue Control Factor (Method1) when RCR = 30Mbps

a. EPRCA



b. EPRCAM



c. ERICA

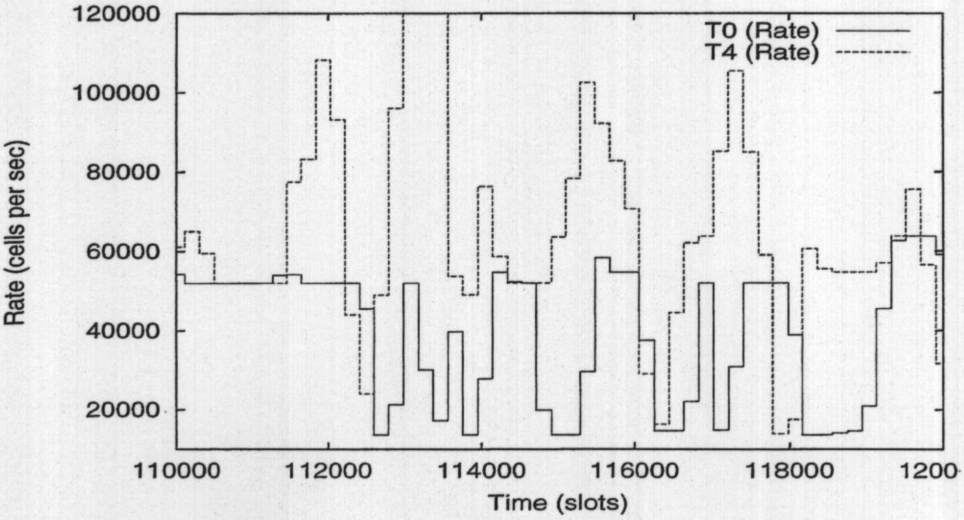


Figure 4.3: Results of ERICA showing unfairness in favor of cross traffic due to high variance in VCShare and Load Factor caused by the Queue Control Factor (Method1) when PCR = 60Mbps

Table 4.1: Results for ERICA for different measurement interval (MI) values
PCR = 30Mbps, Queue Control Factor=Method1

Normalized throughput (MI in μ sec)							
VC	*20*	*40*	*60*	*80*	*100*	*200*	*1000*
T0	0.119	0.123	0.127	0.129	0.130	0.129	0.130
T1	0.119	0.123	0.127	0.129	0.129	0.130	0.130
T2	0.118	0.123	0.127	0.129	0.129	0.130	0.130
T3	0.118	0.123	0.128	0.129	0.129	0.130	0.131
T4	0.151	0.150	0.151	0.151	0.151	0.151	0.147
T5	0.151	0.150	0.151	0.151	0.151	0.150	0.148
T6	0.151	0.149	0.151	0.151	0.151	0.150	0.148
T7	0.159	0.158	0.156	0.154	0.153	0.152	0.149
T8	0.159	0.158	0.156	0.154	0.154	0.153	0.151
T9	0.158	0.158	0.156	0.154	0.154	0.153	0.150
T10	0.164	0.162	0.157	0.155	0.155	0.153	0.150
T11	0.163	0.163	0.157	0.155	0.153	0.153	0.150
T12	0.163	0.162	0.157	0.155	0.155	0.154	0.154
L8	0.928	0.941	0.963	0.968	0.970	0.970	0.964
L12	0.951	0.967	0.977	0.977	0.978	0.977	0.971
L16	0.964	0.978	0.980	0.980	0.980	0.979	0.975
Unfairness (%)							
VC	*20*	*40*	*60*	*80*	*100*	*200*	*1000*
T0	01	00	00	00	01	00	00
T1	01	00	00	00	00	01	00
T2	00	00	00	00	00	01	00
T3	00	00	01	00	00	01	01
T4	28	22	19	17	17	17	13
T5	28	22	19	17	17	16	14
T6	28	21	19	17	17	16	14
T7	35	28	23	19	19	18	15
T8	35	28	23	19	19	19	16
T9	34	28	23	19	19	19	15
T10	39	32	24	20	20	19	15
T11	38	33	24	20	19	19	15
T12	38	32	24	20	20	19	18

Table 4.2: Throughput and Unfairness Results for EPRCA, EPRCAM & ERICA
In ERICA unfairness is due to high variance in VC share caused by Queue Control Factor
Queue Control Factor=Method1, Measurement Interval = 1000 μ sec

Normalized Throughput						
Pcr	*30 Mbps*			*60 Mbps*		
VC	*eprca*	*eprcam*	*erica*	*eprca*	*eprcam*	*erica*
T0	0.131	0.136	0.130	0.116	0.137	0.097
T1	0.132	0.136	0.130	0.118	0.137	0.097
T2	0.132	0.136	0.130	0.117	0.136	0.097
T3	0.131	0.136	0.131	0.115	0.136	0.097
T4	0.145	0.149	0.147	0.145	0.149	0.170
T5	0.145	0.148	0.148	0.145	0.148	0.171
T6	0.145	0.148	0.148	0.147	0.148	0.172
T7	0.147	0.148	0.149	0.151	0.148	0.174
T8	0.148	0.149	0.151	0.150	0.148	0.172
T9	0.147	0.149	0.150	0.149	0.148	0.174
T10	0.147	0.149	0.150	0.151	0.149	0.173
T11	0.148	0.149	0.150	0.151	0.148	0.172
T12	0.148	0.149	0.154	0.153	0.148	0.175
L8	0.962	0.991	0.964	0.903	0.991	0.901
L12	0.968	0.992	0.971	0.916	0.991	0.909
L16	0.968	0.992	0.975	0.920	0.992	0.908
Unfairness (%)						
	All Source Pcr=30 Mbps			*All Source Pcr=60 Mbps*		
VC	*eprca*	*eprcam*	*erica*	*eprca*	*eprcam*	*erica*
T0	00	00	00	01	01	00
T1	01	00	00	03	01	00
T2	01	00	00	02	00	00
T3	00	00	01	00	00	00
T4	11	10	13	26	10	75
T5	11	09	14	26	09	76
T6	11	09	14	28	09	77
T7	12	09	15	31	09	79
T8	13	10	16	30	09	77
T9	12	10	15	30	09	79
T10	12	10	15	31	10	78
T11	13	10	15	31	09	77
T12	13	10	18	33	09	80

Table 4.3: Results of ERICA showing unfairness due to high variance in VC Share caused by Queue Control Factor (Method1, Method2 & Method3) Measurement Interval=1000 μ sec

	Normalized Throughput				Unfairness (%)			
PCR	*30Mbps*	*60Mbps*			*30Mbps*	*60Mbps*		
VC	*Method1*	*Method1*	*Method2*	*Method3*	*Method1*	*Method1*	*Method2*	*Method3*
T0	0.130	0.097	0.133	0.132	00	00	02	00
T1	0.130	0.097	0.132	0.133	00	00	01	01
T2	0.130	0.097	0.132	0.133	00	00	01	01
T3	0.131	0.097	0.131	0.133	01	00	00	01
T4	0.147	0.170	0.150	0.148	13	75	15	12
T5	0.148	0.171	0.147	0.147	14	76	12	11
T6	0.148	0.172	0.152	0.153	14	77	16	16
T7	0.149	0.174	0.150	0.151	15	79	15	14
T8	0.151	0.172	0.150	0.150	16	77	15	14
T9	0.150	0.174	0.151	0.148	15	79	15	12
T10	0.150	0.173	0.149	0.153	15	78	14	16
T11	0.150	0.172	0.153	0.149	15	77	17	13
T12	0.154	0.175	0.151	0.149	18	80	15	13
L8	0.964	0.901	0.977	0.979				
L12	0.971	0.908	0.979	0.980				
L16	0.975	0.908	0.981	0.982				

Table 4.4: Results of ERICA showing unfairness due to high variance in VCShare caused by the Queue Control Factor (Method1)
Case1: Explicit_Rate=Max(FairShare,VCShare), Case2: Explicit_Rate=FairShare
Measurement Interval=1000 μ sec

	Normalized Throughput		Unfairness (%)	
Pcr	*60Mbps*		*60Mbps*	
VC	*Case1*	*Case2*	*Case1*	*Case2*
T0	0.097	0.139	00	1
T1	0.097	0.138	00	0
T2	0.097	0.139	00	1
T3	0.097	0.139	00	1
T4	0.170	0.140	75	1
T5	0.171	0.140	76	1
T6	0.172	0.141	77	2
T7	0.174	0.141	79	2
T8	0.172	0.141	77	2
T9	0.174	0.141	79	2
T10	0.173	0.142	78	3
T11	0.172	0.141	77	2
T12	0.175	0.141	80	2
L8	0.901	0.976		
L12	0.908	0.978		
L16	0.908	0.979		

to above two problems of ERICA. Furthermore, I compared ERICA and EPRCAM in terms of implementation complexity and showed that EPRCAM is much simpler to implement.

Chapter 5

Efficient Use of ABR Bandwidth

5.1 Introduction

The Asynchronous Transfer Mode (ATM) networks are supposed to support a number of services like video, audio, and data transmission. To achieve this objective, ATM defines multiple service classes namely; real time variable bit rate (rt-VBR), non-real time VBR (nrt-VBR), constant bit rate (CBR), available bit rate (ABR), and unspecified bit rate (UBR). The rt-VBR, nrt-VBR and CBR are used for services which need guarantee of quality of service (QoS). However, ABR and UBR are used for data transfer with almost no guarantee of QoS.

The main purpose of the ABR service is to fairly distribute the available bandwidth among contending sources with minimum cell loss. To achieve this objective of ABR service, some congestion control mechanism is necessary. Many congestion control algorithms have been proposed till today. One of the well know and recommended by the ATM forum is the Enhanced Proportional Rate Control Algorithm (EPRCA).

I found that EPRCA has some problems which in some case severely affect its performance in terms of fairness of bandwidth allocation to contending sources. To overcome these problems I proposed some modifications in EPRCA and called the new algorithm as Modified Enhanced Proportional Rate Control Algorithm (EPRCAM) [44].

Though EPRCAM performs very well in terms of throughput and fairness when none or only a small fraction of total link bandwidth is being used by the VBR/CBR sources. However, it is discovered that if VBR/CBR sources stop transmitting and a very large amount of bandwidth becomes available, EPRCAM is unable to utilize the relinquished bandwidth very quickly resulting in very poor throughput and fairness performance.

In this chapter, I propose some changes in the original EPRCAM algorithm in order to make its performance robust even in presence of highly variant VBR/CBR background traffic.

The rest of this chapter is arranged as follows. In section 5.2, I describe the original EPRCAM algorithm. Section 5.3 gives a detailed description of simulation environment. I describe in section 5.4, the problem of EPRCAM which causes poor throughput and fairness performance in highly variant VBR/CBR traffic. The proposed solution of the problem and complete EPRCAM algorithm after proposed modifications is given in section 5.5. I present a comprehensive performance comparison of EPRCA, EPRCAM (original), EPRCAM (after changes) and ERICA algorithms in section 5.6. Note that ERICA in this chapter refers to ERICA+ algorithm. This algorithm is a well known congestion notification algorithm recommended by the ATM forum. Finally I summarize and conclude this chapter in section 5.7.

5.2 Modified Enhanced Proportional Rate Control Algorithm (EPRCAM)

The Enhanced Proportional Rate Control Algorithm (EPRCA) is one of the recommended algorithms for congestion control by the ATM forum [48]. I identified that due to no control by the switch in the no congestion state in EPRCA, unfairness is caused in bandwidth utilization by the contending ABR sources. I proposed some modifications in EPRCA and called it Modified EPRCA (EPRCAM) [44]. Below is the complete description of EPRCAM algorithm.

Like EPRCA, in EPRCAM a switch maintains three congestion states namely *no congestion* (NCONG), *low congestion* (LCONG) and *high congestion* (HCONG) state. The congestion states are decided according to the following rules. These rules use two thresholds called a *low threshold* (LTH) and a *high threshold* (HTH) on the length of queue in the switch (SWQ).

```
IF SWQ>=HTH THEN HCONG=TRUE
IF SWQ=LTH AND SWQ<HTH) THEN LCONG=TRUE
```

A switch also maintains an exponential moving average of CCR (Current Cell Rate) values of sources taken from the Forward Resource Management cells (FRM) of the source virtual connections (VCs) passing through it. This exponential moving average is called Mean of Allowed Cell Rates (MACR) and is calculated according to the following expression.

```
MACR=(1-a)*MACR+a*CCR
```

On arrival of an BRM cell, a switch calculates explicit rate (ER) for the source to which the BRM cell belongs. The ER is calculated according to the following rules.

```
IF (HCONG) AND (ER>=MACR*MRF) THEN
        ER = Min(ER,MACR*MRF)

IF (LCONG) AND (ER>=MACR*DPF) THEN
        ER = Min(ER,MACR*ERF)

IF (NCONG) THEN
       ER = Min(ER,MACR*(1+RIFS))

Where, MRF = Major Reduction Factor
       ERF = Explicit Reduction Factor
       DPF = Down Pressure Factor
```

The recommended values of "a", MRF, ERF and DPF are the same as those used by EPRCA as given in [48].

The RIF factor at the source is set to 1 in order for the source to always set its ACR to the ER value of the returning RM cells. Moreover, a value of 1/16 is recommended for RIFS in [44].

5.3 Network Simulation Model

5.3.1 Description of Model & Parameter Setting

In order to observe the performance of EPRCAM in presence of highly variant VBR/CBR background traffic, computer simulations were performed using a network model as shown in Figure 5.1. This model consists of a wide area network with four output buffer type switches (4x4), thirteen sources and three destinations. The output buffers in each switch are infinite size, and are donated by Q0, Q1, Q2 and Q3. The sources are divided into four groups. The dashed lines show the traffic flow.

Group-A: Source T0 to T3 (end-to-end traffic)
Group-B: Source T4 to T6 (two-hop traffic)
Group-C: Source T7 to T9 (two-hop traffic)
Group-D: Source T10 to T12 (one-hop traffic)

The values of parameters used are given below. Other parameters are taken as recommended by ATM Forum [48].

- low threshold = 20 cells
- high threshold = 100 cells
- link capacity = 155.52 Mbps
- PCR of each source = 60 Mbps
- each link = 120 Km
- simulation time = 1.00 sec
- source model = persistent sources
- VBR/CBR transmission duration = From time slot 30,000 to time slot 60,000

By persistent sources I mean that if allowed by the allowed cell rate (ACR) values, sources have all the time cell to transmit up to the ACR values.

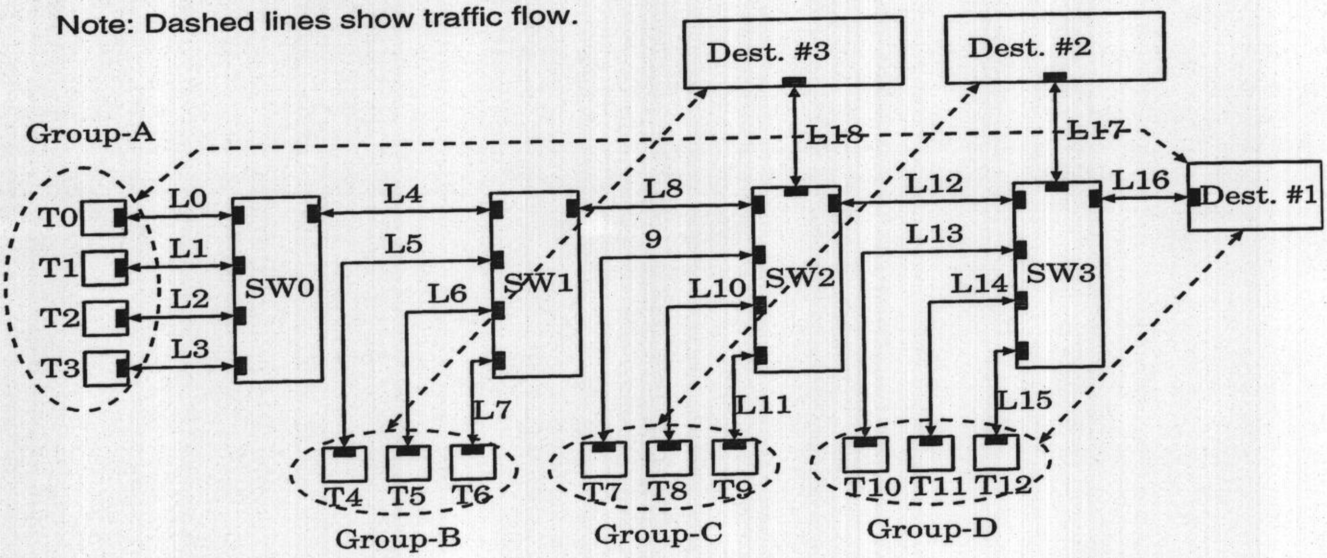

Note: Dashed lines show traffic flow.

Figure 5.1: Network Simulation Model

These simulations were carried out for the following two cases of background traffic. For the convenience of simulation it is assumed that each VBR/CBR source utilizes a bandwidth of 64Mbps.

Case1: Only two sources (T0 and T1) of the end-to-end traffic are VBR/CBR sources and the rest are ABR sources.

Case2: In this case one source each from the end-to-end and cross traffic (T0, T4, T7, and T10) is the VBR/CBR source and the rest are ABR sources.

Note that the load created by the VBR/CBR traffic at each switch except switch Sw0 in each case is the same that is, 128 Mbps of total link capacity of 155 Mbps. These cases are designed in such a way so as to evaluate the performance of a congestion control algorithm in case when a large amount of bandwidth becomes unavailable/available for ABR traffic.

5.3.2 Fairness Definition

Since a strict definition of unfairness is not available, our own definition of unfairness is given by.

$$F_n(\%) = (T_n - T_i)/T_i * 100$$

such that $T_i \leq T_j$ For $j = 0, 1, ..., N$
where, F_n = Unfairness in % for source n,
and, T_n = Normalized throughput of source n.

In the above expression, $F_n(\%)$ gives the difference between the throughput of a source and that of the source which achieved minimum throughput. The condition in the "where" clause states that the fairness comparison can only be done between sources which expect same throughput (according to max-min criteria in [48]).

5.4 Problem Identification

In [44], I identified that a switch which uses EPRCA for congestion control notifies its sources the rate it can supports in case the switch is congested. However, when the switch is not congested, it does not provide any notification to its sources and thus sources keep on increasing their rates of transmission b a factor of their PCR values till the network becomes congested again. We showed that no congestion control in no congestion state leads to unfairness due to possible diverse source rates. This approach also requires proper tuning of source parameters like RIF, RDF and PCR.

I proposed that even in the no congestion state a switch should notify its sources of the explicit rate of transmission which should depend on MACR (section 5.2). I showed through simulation results that the fairness and throughput performance of EPRCAM is much better than that of EPRCA. Moreover, it did not require tuning of source parameters of RIF, RDF and PCR.

Unfortunately EPRCAM was not evaluated in an environment where large amount of bandwidth becomes unavailable/available very quickly due to background VBR/CBR traffic. I have found that in such a scenario, performance of EPRCAM in terms of throughput becomes extremely poor. The fairness performance, however, may also become poor. The reason behind this poor performance is as given below.

5.4.1 Poor Bandwidth Utilization

When a large amount of bandwidth becomes available, MACR which is the average of current cell rates of all sources passing through a switch is slow to react due to its nature of exponential averaging. Depending on the amount of the bandwidth which has become available, it might take a large number of round trips before MACR and the source rates become adjusted. Of course, during all those round trips, the network is being unnecessarily underutilized. The underutilization depends on the amount of bandwidth that had become instantly available and the MACR value at that instant.

5.4.2 Poor Fairness Performance

When VBR/CBR traffic is over and a large amount of bandwidth becomes available, sources are notified to increase their rates by a factor of MACR value of the most congested switch in their path. If *different round trip delay VBR/CBR sources* at different switches stop transmitting, MACR values at different switches become diversified which leads to diversified source rates and obviously results in unfairness. Remember that the source rates and MACR values are interdependent. This unfairness continues till, source rates are increased to a point where all of the available bandwidth becomes utilized and the network becomes congested once again. At this point source rates and MACR values at different switches become adjusted.

5.5 Proposed Solution

The reason behind the underutilization and fairness problem of network when a large amount of bandwidth becomes unavailable/available due to background VBR/CBR traffic (refer section 5.4) is that EPRCAM has no mechanism to judge the exact amount of available bandwidth and thus can not notify an optimal rate to sources quickly. Instead it keeps on increasing source rates slowly (depending on MACR value) until the network becomes fully utilized. At this point, network becomes congested and the sources with high transmission rates are asked to reduce their rates and those with low transmission rates are asked to increase their rates. We can see that the main reason behind low throughput and unfairness is the slow reactiveness of MACR to judge network utilization.

Now that we know the main cause of the problem I suggest that, when computing explicit rate, EPRCAM should incorporate the load factor (LF) to judge the available bandwidth along with the queue factor (QF) and MACR to avoid cell loss. Hence, the explicit rate (ER) in EPRCAM can be calculated using the following expression.

```
ER = MACR*QF/LF
```

To avoid frequent changes in load factor LF and thus in ER, exponential averaging of LF is used. To calculate LF, the number of cells counted (CELL_COUNT) in a measurement interval (MI) is divided by MI.

```
NLF = CELL_COUNT/MI
LF  = LF+(NLF-LF)*AF
```

Where, AF is the averaging factor and I recommend a value of 1/4 for AF. Note that LF greater than 1 represents overload and LF less than 1 represent underload. The queue factor (QF) is calculated according to the following expressions which are similar to the original EPRCAM except a few minor changes.

```
IF(HCONG AND LF>1) THEN QF=MRF ---->(1)
IF(LCONG)          THEN QF=ERF ---->(2)
IF(NCONG AND LF<1) THEN QF=1+RIFS-->(3)
```

For recommended values of MRF and ERF please refer [48]. For RIFS see [44]. For HCONG, LCONG, and NCONG please refer section 5.2.

In (1), when the switch is in a high congestion (HCONG) state and the LF is greater than 1 then QF is set to major reduction factor (MRF). The reason behind this is that while queue may still be over high threshold, the load at the input links may be diminishing due to earlier notifications for major reduction. In that case another major reduction may result in unnecessary underutilization.

In (2), if low congestion (LCONG) is observed, QF is set to explicit reduction factor (ERF).

In expression (3), when there is no congestion (NCONG) at a switch and there is underload, QF is set to 1+RIFS. In case of no congestion and an overload there is no need for rate increase which may result in unnecessary congestion at the switch.

5.5.1 Impact of Measurement Interval

It is expected that the impact of the length of measurement interval on the performance will be minimal because the measurement interval is used only for measuring input load. Moreover, the measured load has only a weight of 25% in the overall load. However, small measurement interval will result in quick changes in the overload. Since, cross traffic is quick to react to any changes, this may result in some unfairness in favor of cross traffic. I will evaluate the impact of the length of measurement interval and recommend its value in section 5.6.4.

5.5.2 Implementation Complexity

Regarding the implementation complexity of the EPRCAM algorithm after the changes proposed in this section, its complexity will increase. This is because the switch has to compute load factor (LF) and, moreover, LF is also being taken into account during calculation of explicit rate (ER).

5.6 Performance Comparison

For the sake of discussion, I denote EPRCAM after the proposed changes as EPRCAM'. I discuss in subsequent sections the performance of EPRCAM, EPRCAM', EPRCA and ERICA. The ERICA is also a well known explicit congestion notification algorithm.

Case1: To compare performance of EPRCAM, EPRCAM', EPRCA and ERICA algorithms when a large amount of bandwidth becomes unavailable/available due to VBR/CBR traffic, a simulation was performed in which sources T0 and T1 (end-to-end traffic) are VBR/CBR sources and the rest are ABR sources. Each VBR/CBR source transmits at a rate of 64 Mbps from time slot 30,000 to time slot 60,000 of the total simulation time of 1.00 sec which is from 0 to around 100,000 time slots. I refer to this scenario as case1. Total VBR/CBR load at each switch is 128 Mbps.

Case2: To further clarify the performance of these algorithms another pattern of VBR/CBR traffic is selected in which one source from each group of sources is selected to be a VBR/CBR source (T0, T4, T7, and T10) making a VBR/CBR load of 128 Mbps at each switch which is the same as in case1. I refer to this scenario as case2.

5.6.1 Queue Length

Since the load at all switches is the same except Sw0 which is not congested at all, buffer occupation of only Sw1 is shown to observe the queue fill up pattern due to highly variant VBR/CBR traffic.

Note that in all graphs, buffer occupation is shown in terms of number of cells.

Case1: Figures 5.2a, 5.2b, 5.3a, and 5.3b show buffer occupation of switch Sw1, queue Q0 for EPRCAM, EPRCAM', EPRCA and ERICA for case1 (section 5.3.1) respectively. We can see from these figures that for each algorithm, when the VBR/CBR sources start transmitting at around time slot 30,000, queues explode and reach values between 500 to 600 cells.

Case2: The corresponding buffer occupations for case2 are shown in Figures 5.6a, 5.6b, 5.7a and 5.7b . These figures show that for each algorithm, when the VBR/CBR sources start transmitting at around time slot 30,000, queues reach values between 600 to 700 cells.

Discussion: A point worth noting here is that in case1 & case2, when VBR/CBR traffic stops transmitting cells at around time slot 60,000, switch queues for EPRCAM become almost completely empty implying underutilization. On the other hand for EPRCAM', switch queues once again shoot to around 400 cells due to its attempt to quickly utilize the available bandwidth. However, note that due to queue control factor, queues still remain under control and overall buffer requirement does not increase.

5.6.2 Throughput

In this section we compare performance of different congestion control algorithms in terms of link utilization and throughput of sources. The throughput of sources is compared by analyzing their current cell rate curves which ultimately determine the total throughput achieved by the sources.

Case1: The rates of Group-A (end-to-end traffic) for EPRCAM, EPRCAM', EPRCA and ERICA for case1 are shown in Figures 5.4a, 5.4b, 5.5a and 5.5b .

We can observe that when the VBR/CBR sources start transmitting at around time slot 30,000, the rates of sources T2 & T3 become substantially reduced. While the VBR/CBR traffic is transmitting, the rates of ABR sources remain reduced and fluctuate at a very low level.

At the time slot 60,000, VBR/CBR sources stop transmitting and thus 128 Mbps (T0=T1=64Mb becomes available instantly. Here, we can observe from Figure 5.2a that in EPRCAM the ABR sources are unable to quickly increase their rates but only slowly keep on increasing for the remaining period of the simulation indicating substantial wastage of precious link bandwidth.

On the other hand, we see from Figure 5.2b for EPRCAM' that when VBR/CBR sources quit transmitting cells, the ABR source rates are quickly increased depending on the amount of available bandwidth and queue size. We can observe that for a slight duration the rates even overshoot but are quickly brought to the optimal value that can be supported by the network.

In case of EPRCA, the rate curves (Figure 5.3a) show increments in steps. The size of these steps depends on the PCR values of the sources. Hence, the recovery period depends on the source PCRs. Regarding ERICA, the recovery is very quick, as can be observed from Figure 5.3b.

Case2: The rates of Group-A (end-to-end traffic) for EPRCAM, EPRCAM', EPRCA and ERICA for case2 are shown in Figures 5.8a, 5.8b, 5.9a and 5.8b. These figures show the same kind of behavior as observed in case1.

Discussion:

To observe the overall performance of each source and the congested links (L8, L12, L16) for the EPRCAM, EPRCAM', EPRCA, and ERICA, throughput results are summarized in Tables 5.1 and 5.2 for case1 and case2 respectively. Each table gives throughput and corresponding unfairness performance in favor of cross traffic for three portions of the simulation duration that is;

1. The duration of the simulation before VBR/CBR sources started their transmission. The results for this duration are given under the caption of *"Before VBR/CBR"*.

2. The time period of the simulation when VBR/CBR sources are active. The caption is *"During VBR/CBR"*.

3. And the time period of the simulation after the VBR/CBR sources stopped their transmission. The results are given under the caption *"After VBR/CBR"*.

Looking at *eprcam* column of Table 5.1 (case1), we see that the link utilization by EPRCAM is around 94% and 98% before and during VBR/CBR traffic respectively, however, the links are being utilized at around only 44% after VBR/CBR which is a very heavy loss (50%) of throughput. The same can be observed from Table 5.2 (case2). Here, the link utilization after VBR/CBR, has even dropped to a maximum of only 32%. On the other hand, if we look at the column *eprcam'* of Table 5.1, we observe that the link utilization for EPRCAM' is around 97% before and during VBR/CBR traffic respectively and 92% after VBR/CBR. The same pattern of results can be seen under column *eprcam'* in Table 5.2.

This clearly shows that the performance of EPRCAM after proposed changes has been substantially improved .

5.6.3 Fairness

Case1: As already described in section 5.6.2, Table 5.1 gives throughput as well as unfairness results for EPRCAM, EPRCAM', EPRCA and ERICA for case1.

Looking at the column *eprcam* and *eprcam'*, we can see that for both EPRCAM and EPRCAM', the unfairness in favor of cross traffic is under 10% before, during and after VBR/CBR traffic.

Case2: Corresponding unfairness results for case2 are given in Table 5.2.

We observe that for EPRCAM' the unfairness is around 10% before and after VBR/CBR traffic, but a maximum of 35% during VBR/CBR traffic. On the other hand, looking at the column *eprcam*, we see that for EPRCAM, the unfairness has become very high that is during VBR/CBR traffic, it is a maximum of 99% and after VBR/CBR a maximum of 286% in favor of cross traffic.

Discussion:

The fairness performance degradation in EPRCAM is mainly due to the reason that when VBR/CBR sources stop transmitting cells, the MACR values at different switches become diversified if there are VBR/CBR sources of different round trip delay at different switches. Diversified MACR values lead to diversified source rates. The source rates keep on increasing till the network becomes congested once again. At this point MACR values in different switches in the network become adjusted. That is why we observe that in case1, though, link utilization is very poor but unfairness after VBR/CBR sources stop transmitting remains low (maximum of 10%). Note that round trip delay of each VBR/CBR source (end-to-end traffic) at each switch is the same. On the other hand in case2, we see that due to different round trip delay VBR/CBR traffic at different switches, unfairness in favor of cross traffic reaches a maximum of 286%.

In a switch with EPRCAM', impact of different round trip delay VBR/CBR sources is minimum due to the reason that as soon as a large amount of bandwidth becomes available, the source rates are increased accordingly and network becomes congested. Hence, even if MACR values at different switches were diversified, they become quickly adjusted.

Similarly, comparing the unfairness of *eprcam'*, *eprca*, and *erica* for case1 & case2 in Table 5.1 and 5.2, we see that the unfairness performance of EPRCAM' before, during and after VBR/CBR traffic is better than that of EPRCA and ERICA. However the performance of EPRCAM is worse than that of EPRCA and ERICA during and after VBR/CBR traffic.

5.6.4 Impact of Measurement Interval

To clarify, the impact of measurement interval on the performance of EPRCAM', simulations for case1 and case2 are repeated for three different values of the interval that is, 1 millisec, 1/10 millisec, and 1/100 millisec.

Case1: Throughput and fairness results for case1 are given in Table 5.3. Each column heading represents the length of interval.

From Table 5.3, we can see that reducing measurement interval 10 times, that is from 1 millisec to 1/10 millisec, increases total link throughput by only 1%. Further reducing it to 1/100 millisecs does not increase link throughput at all. Hence smaller values of interval give only negligible throughput gain.

On the other hand the *unfairness* results in the same table show that reducing interval from 1 millisec to 1/10 millisec, almost doubles the unfairness in favor of cross traffic from around 5% to 10%. Similarly, further reducing interval from 1/10 millisec to 1/100 millisec, once again doubles unfairness in favor of cross traffic from around 10% to 25%.

Case2: Corresponding throughput and fairness results for case2 are given in Table 5.4. It can be observed that the impact of measurement interval in case2 is similar to that of case1.

Discussion: The reason behind the unfairness caused due to the reduction in measurement interval is that when interval is reduced, changes in overload are made more frequently. The cross traffic due to its shorter round trip delay is always quick to react. Hence, variance in load factor gives rise to unfairness in favor of cross traffic.

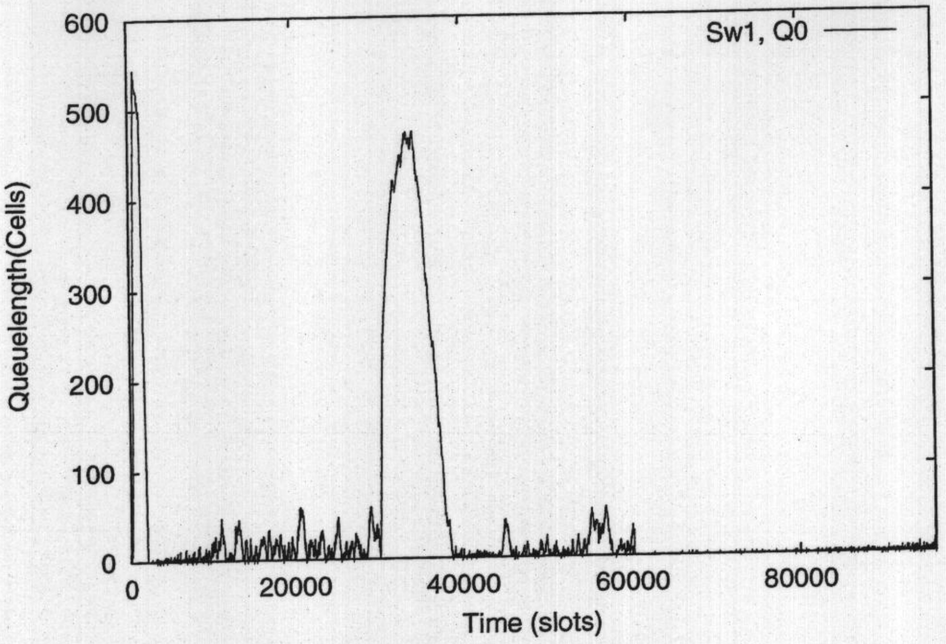
We have seen that the interval of 1 millisec for load factor measurement has produced the best results in all of the situations in our simulation model. Though it is difficult to generalize this value for other network models with different link speeds, we expect that this value should perform fairly well in networks with link speeds of 155.52 Mbps. At present we are analyzing the impact of the interval on other network models with different link speeds.

5.7 Summary & Conclusion

In this chapter, I first described EPRCAM algorithm which is claimed to have solved the unfairness problem of EPRCA algorithm. I then identified that the throughput and fairness performance of EPRCAM depends on the MACR (Mean of Allowed Cell Rates) and in case if a large amount of bandwidth becomes unavailable/available suddenly due to VBR/CBR traffic, EPRCAM is unable to quickly utilize this bandwidth due to the slow reactiveness of MACR. It also produces unfairness in bandwidth allocation to the contending ABR sources.

I suggested that the reason behind this is that EPRCAM is unable to exactly estimate the available bandwidth. I proposed that in EPRCAM, besides queue factor and MACR, load factor should also be taken into account when notifying sources of explicit rate to quickly and fairly distribute available bandwidth among contending ABR sources. I showed through simulation results that this approach overcomes the problems stated above and results in much improved throughput and fairness performance.

a. EPRCAM



b. EPRCAM' (Measurement Interval=.001 sec)

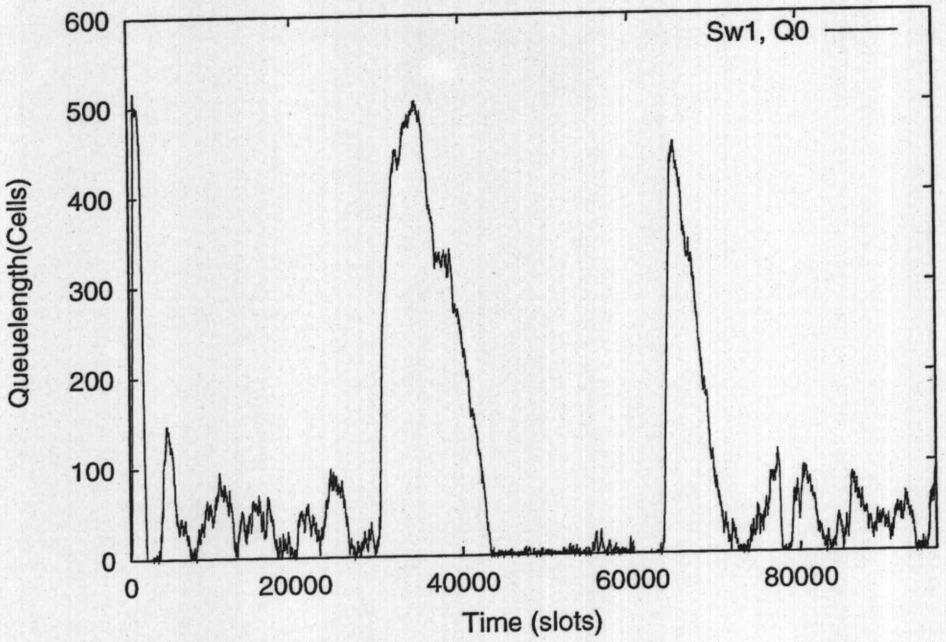
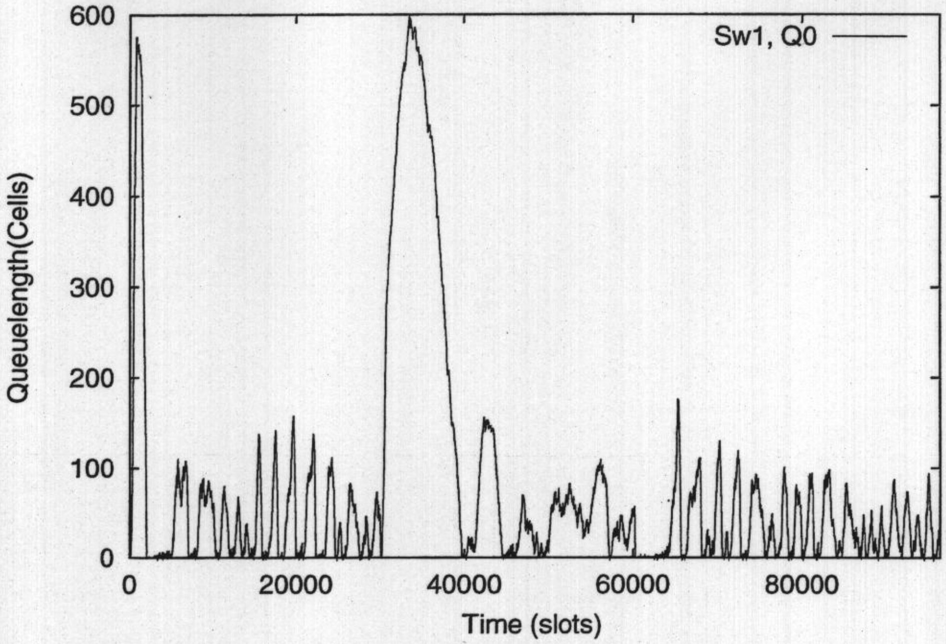


Figure 5.2: Comparison of EPRCAM, EPRCAM' regarding buffer requirement due to background VBR/CBR (Case1).

a. EPRCA



b. ERICA

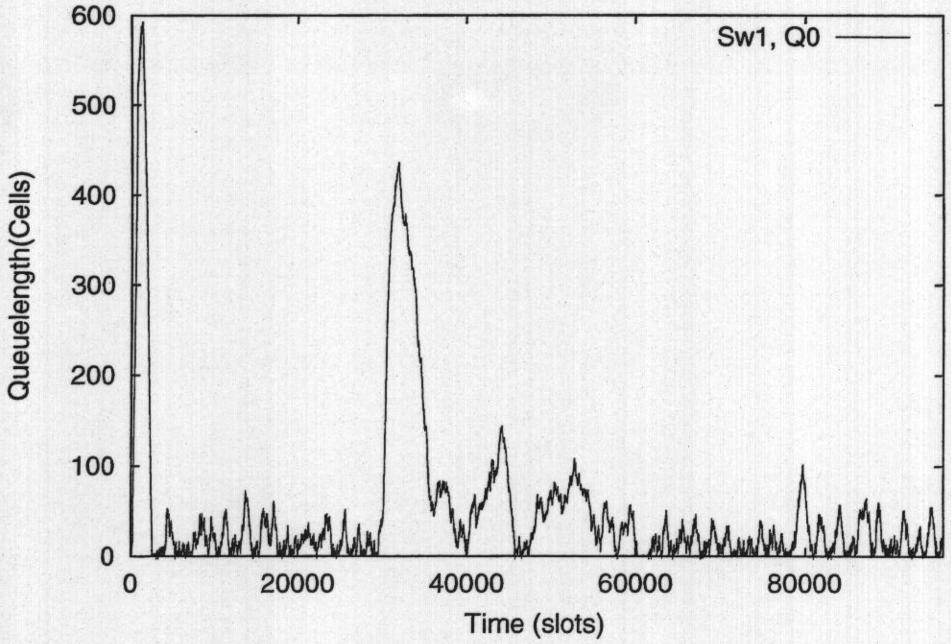
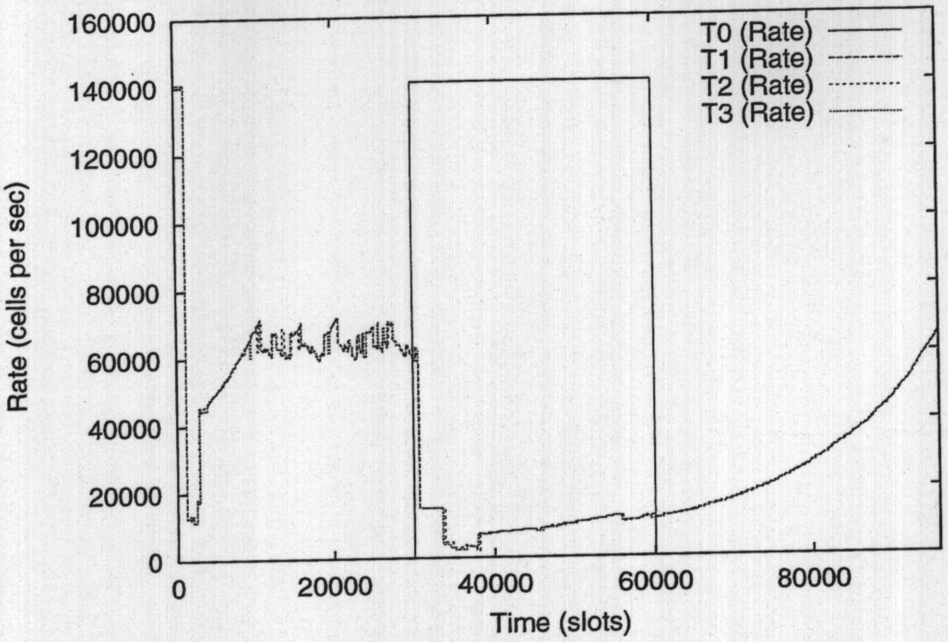


Figure 5.3: Comparison of EPRCA & ERICA, regarding buffer requirement due to background VBR/CBR (Case1).

a. EPRCAM



b. EPRCAM' (Measurement Interval=.001 sec)

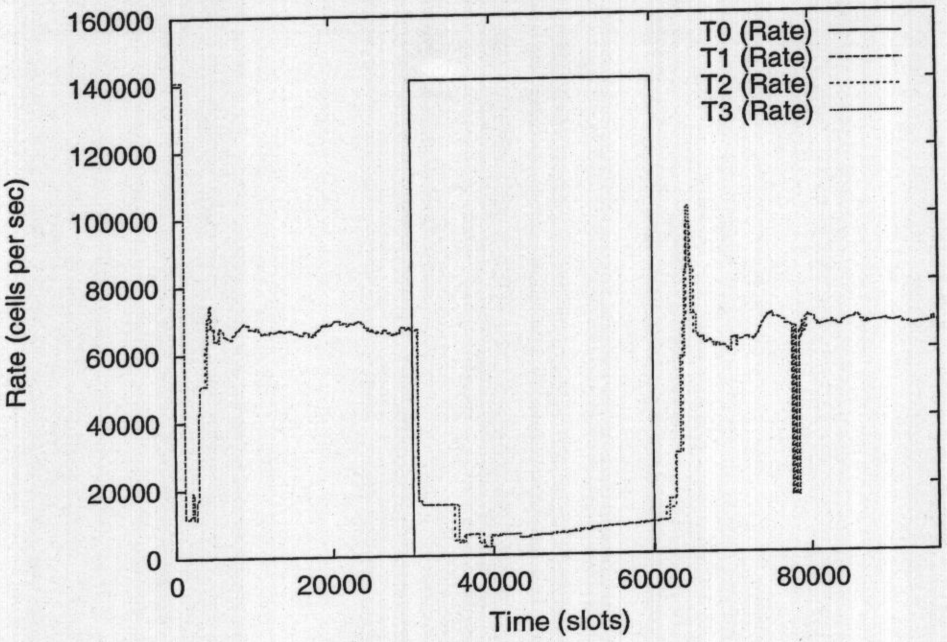
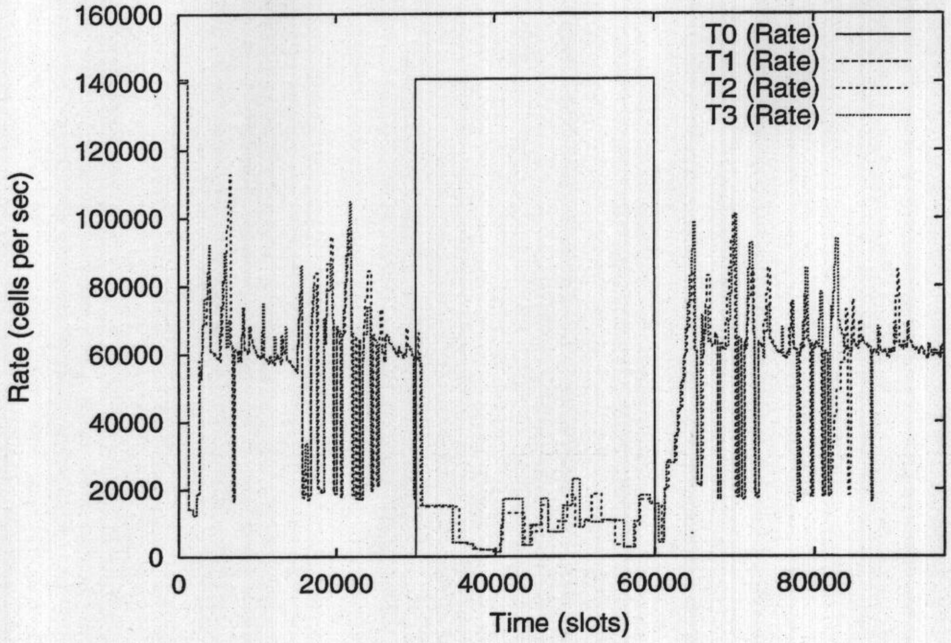


Figure 5.4: Comparison of EPRCAM, EPRCAM' in terms of flow of source rates due to background VBR/CBR (Case1).

a. EPRCA



b. ERICA

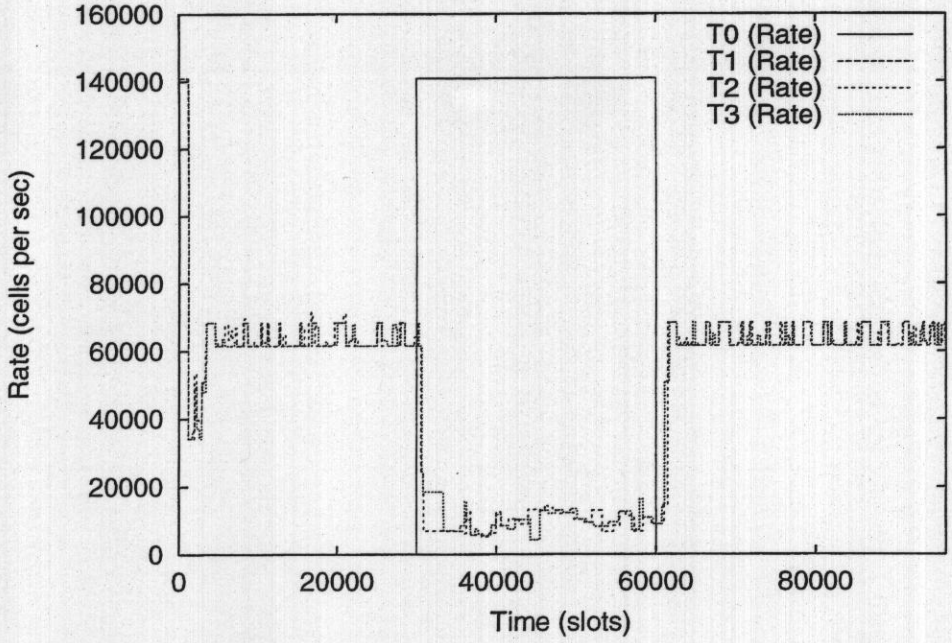
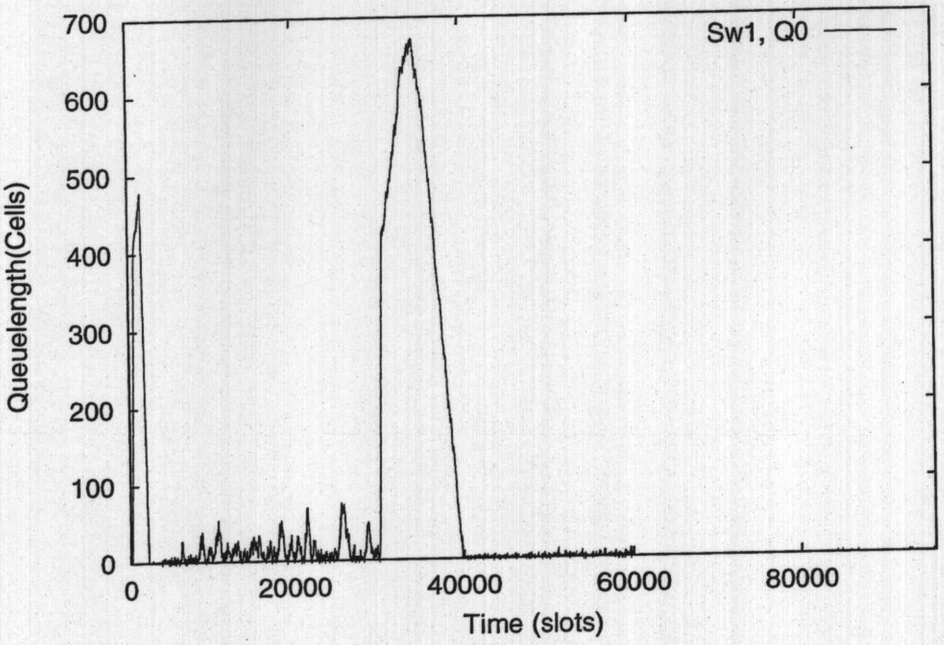


Figure 5.5: Comparison of EPRCA & ERICA, in terms of flow of source rates due to background VBR/CBR (Case1).

a. EPRCAM



b. EPRCAM' (Measurement Interval=.001 sec)

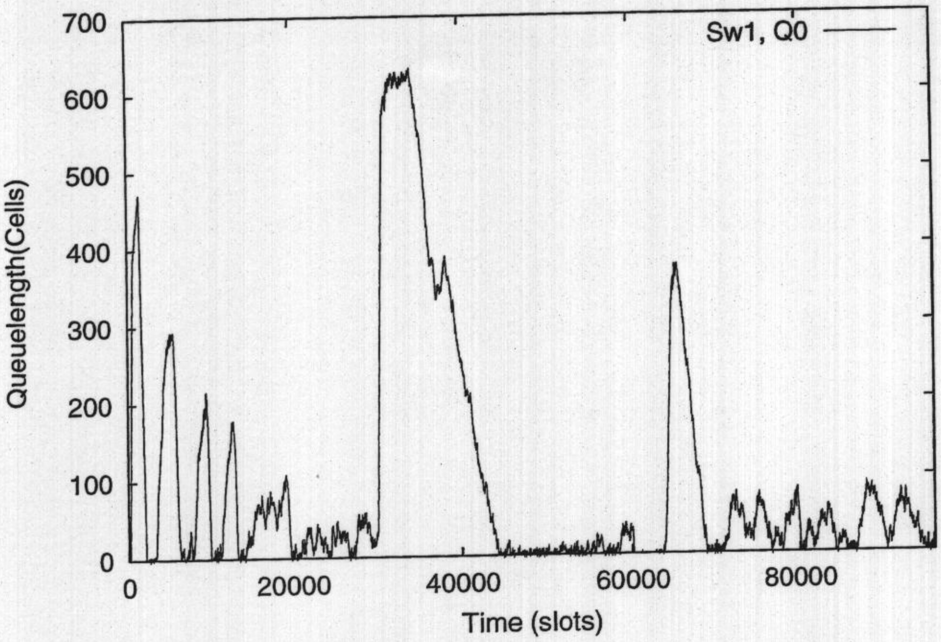


Figure 5.6: Comparison of EPRCAM, EPRCAM' regarding buffer requirement due to background VBR/CBR (Case2).

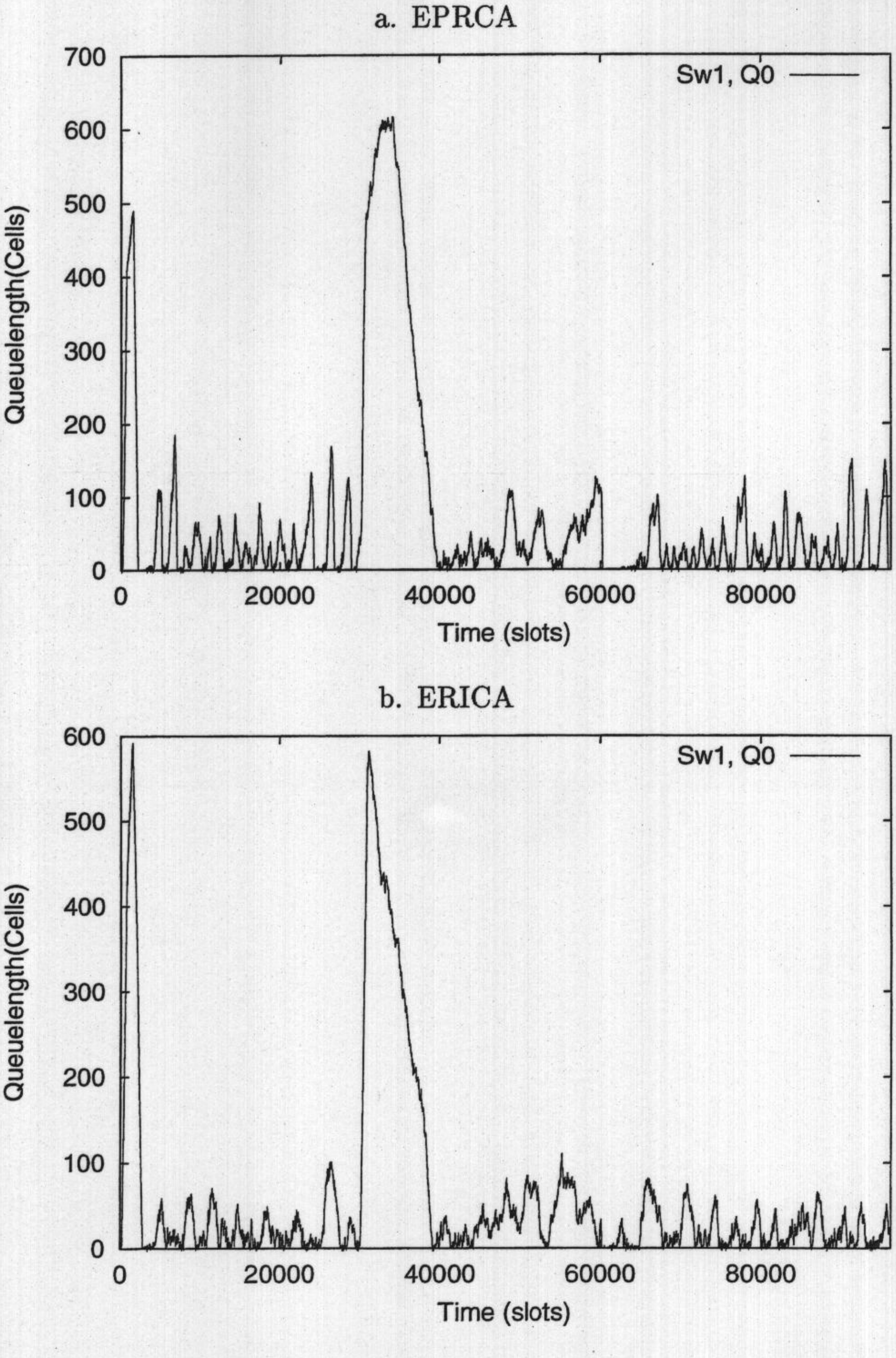
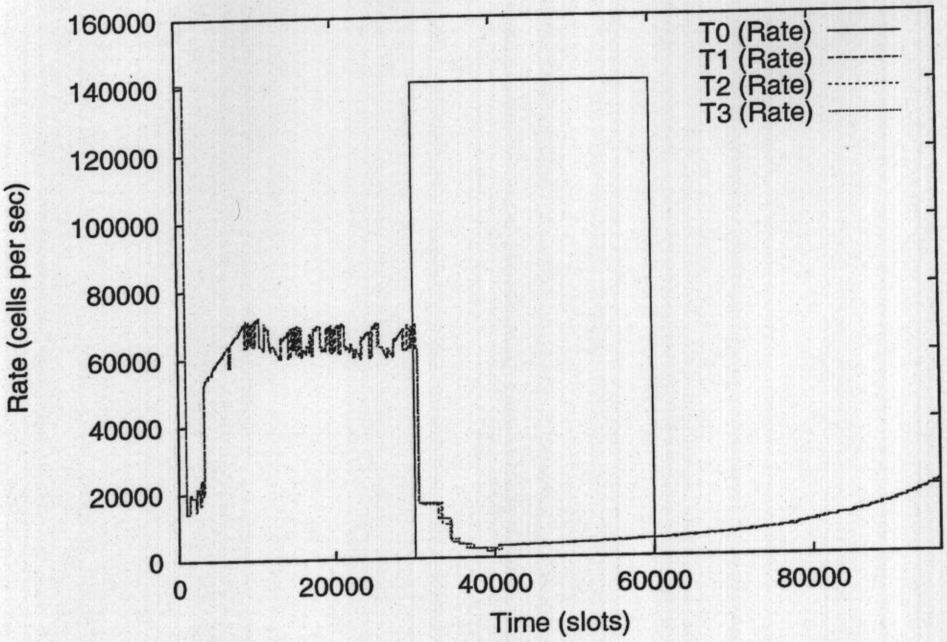


Figure 5.7: Comparison of EPRCA & ERICA, regarding buffer requirement due to background VBR/CBR (Case2).

a. EPRCAM



b. EPRCAM' (Measurement Interval=.001 sec)

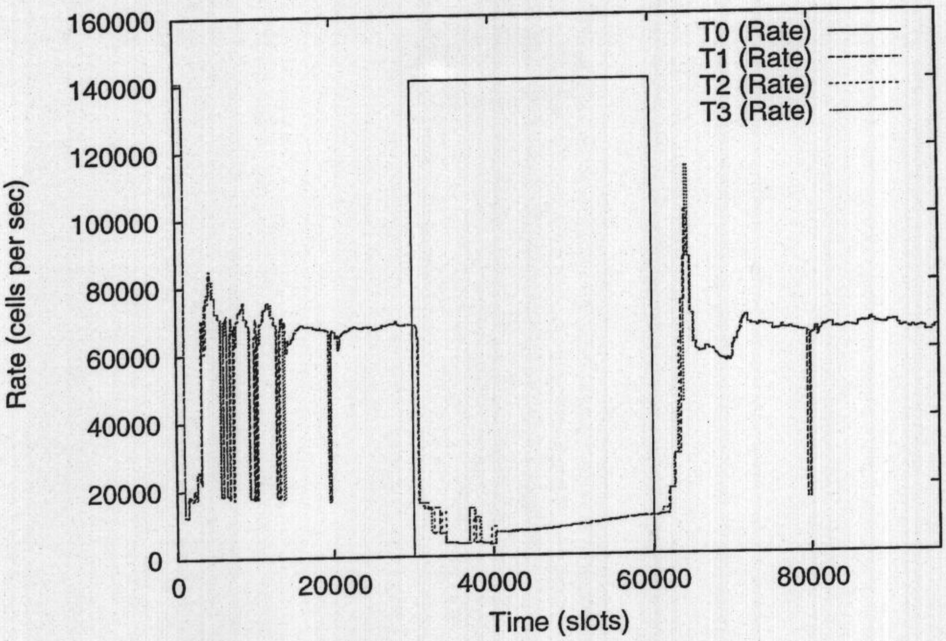
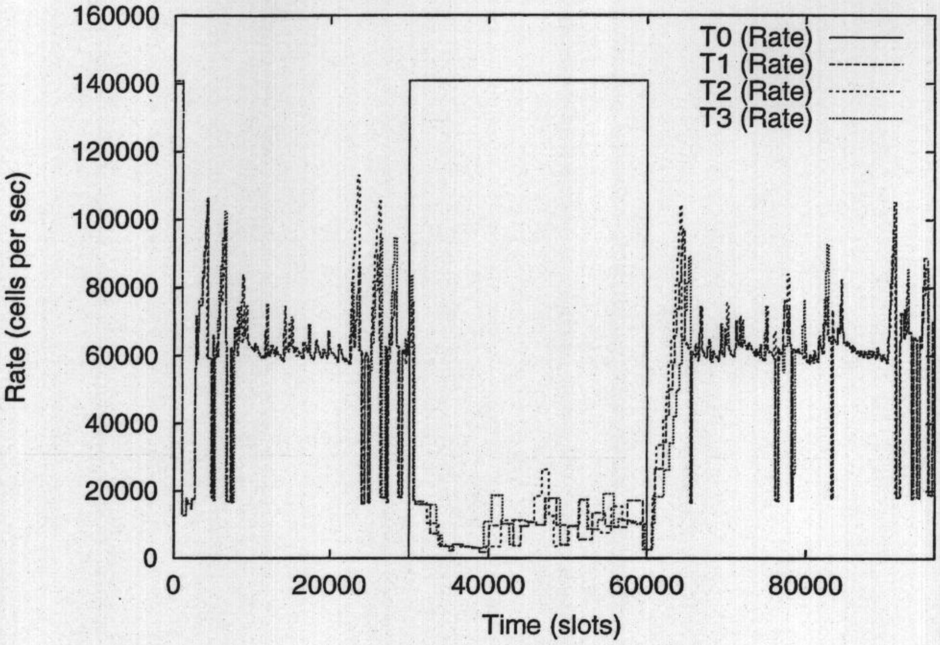


Figure 5.8: Comparison of EPRCAM, EPRCAM' in terms of flow of source rates due to background VBR/CBR (Case2).

a. EPRCA



b. ERICA

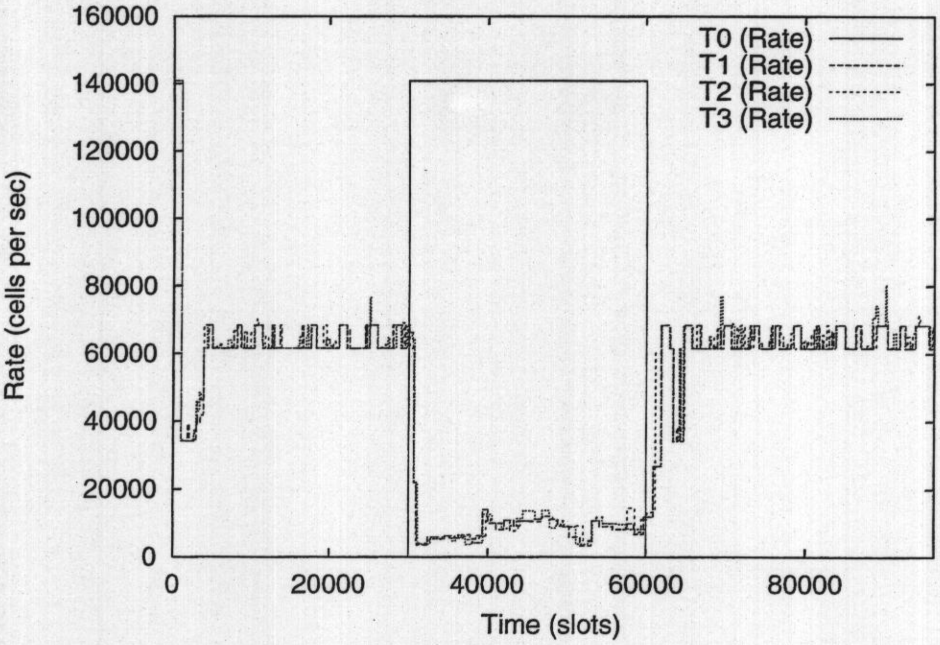


Figure 5.9: Comparison of EPRCA & ERICA, in terms of flow of source rates due to background VBR/CBR (Case2).

Table 5.1: Throughput & Unfairness Results Showing Impact of VBR/CBR (Case1: T0=T1=64Mbps, Interval for EPRCAM'=.001 sec)

	Throughput				*Unfairness*			
	Before VBR/CBR							
VC	*eprcam*	*eprcam'*	*eprca*	*erica*	*eprcam*	*eprcam'*	*eprca*	*erica*
T2	0.181	0.191	0.174	0.187	1	2	1	0
T3	0.180	0.188	0.172	0.188	0	0	0	1
T4	0.186	0.192	0.176	0.199	3	2	2	6
T5	0.181	0.193	0.182	0.187	1	3	6	0
T6	0.182	0.195	0.197	0.202	1	4	15	8
T7	0.181	0.194	0.184	0.202	1	3	7	8
T8	0.186	0.197	0.196	0.195	3	5	14	4
T9	0.183	0.192	0.187	0.194	2	2	9	4
T10	0.190	0.199	0.200	0.200	6	6	16	7
T11	0.195	0.194	0.202	0.201	8	3	17	7
T12	0.197	0.200	0.196	0.202	9	6	14	8
L8	0.911	0.959	0.902	0.963				
L12	0.911	0.962	0.914	0.967				
L16	0.943	0.972	0.944	0.978				
	During VBR/CBR							
VC	*eprcam*	*eprcam'*	*eprca*	*erica*	*eprcam*	*eprcam'*	*eprca*	*erica*
T2	0.033	0.029	0.157	0.033	6	0	0	0
T3	0.034	0.029	0.168	0.036	10	0	7	9
T4	0.033	0.033	0.201	0.051	6	14	28	55
T5	0.031	0.032	0.186	0.037	0	10	18	12
T6	0.034	0.029	0.193	0.035	10	0	23	6
T7	0.032	0.033	0.189	0.037	3	14	20	12
T8	0.032	0.032	0.197	0.047	3	10	25	42
T9	0.032	0.032	0.191	0.040	3	10	22	21
T10	0.034	0.034	0.203	0.041	10	17	29	24
T11	0.033	0.030	0.198	0.048	6	3	26	45
T12	0.032	0.031	0.195	0.036	3	7	24	9
L8	0.971	0.965	0.989	0.996				
L12	0.968	0.968	0.991	0.997				
L16	0.972	0.967	0.993	0.998				
	After VBR/CBR							
VC	*eprcam*	*eprcam'*	*eprca*	*erica*	*eprcam*	*eprcam'*	*eprca*	*erica*
T2	0.084	0.174	0.157	0.178	2	0	0	0
T3	0.082	0.177	0.168	0.180	0	2	7	1
T4	0.088	0.186	0.201	0.191	7	7	28	7
T5	0.087	0.188	0.186	0.208	6	8	18	17
T6	0.086	0.187	0.193	0.188	5	7	23	6
T7	0.084	0.184	0.189	0.192	2	6	20	8
T8	0.085	0.184	0.197	0.196	4	6	25	10
T9	0.086	0.187	0.191	0.208	5	7	22	17
T10	0.084	0.186	0.203	0.192	2	7	29	8
T11	0.087	0.191	0.198	0.197	6	9	26	11
T12	0.086	0.183	0.195	0.201	5	5	24	13
L8	0.440	0.923	0.915	0.957				
L12	0.433	0.918	0.913	0.965				
L16	0.435	0.923	0.932	0.960				

Table 5.2: Throughput & Unfairness Results Showing Impact of VBR/CBR (Case2: T0=T4=T7=T10=64Mbps, Interval for EPRCAM'=.001 sec)

	Throughput				*Unfairness*			
	Before VBR/CBR							
VC	*eprcam*	*eprcam'*	*eprca*	*erica*	*eprcam*	*eprcam'*	*eprca*	*erica*
T1	0.179	0.184	0.176	0.179	0	2	0	0
T2	0.184	0.182	0.179	0.182	3	1	2	2
T3	0.190	0.181	0.177	0.183	6	0	1	2
T5	0.187	0.198	0.191	0.192	4	9	9	7
T6	0.182	0.199	0.186	0.220	2	10	6	23
T8	0.190	0.201	0.199	0.212	6	11	13	18
T9	0.194	0.204	0.194	0.210	8	13	10	17
T11	0.199	0.205	0.202	0.201	11	13	15	12
T12	0.194	0.212	0.205	0.220	8	17	16	23
L8	0.921	0.944	0.910	0.957				
L12	0.936	0.952	0.926	0.967				
L16	0.946	0.965	0.939	0.966				
	During VBR/CBR							
VC	*eprcam*	*eprcam'*	*eprca*	*erica*	*eprcam*	*eprcam'*	*eprca*	*erica*
T1	0.022	0.031	0.030	0.030	10	0	0	0
T2	0.021	0.032	0.031	0.032	5	3	3	7
T3	0.020	0.032	0.035	0.032	0	3	17	7
T5	0.022	0.034	0.040	0.050	10	10	33	67
T6	0.021	0.035	0.040	0.036	5	13	33	20
T8	0.039	0.040	0.040	0.047	95	29	33	57
T9	0.040	0.042	0.038	0.044	99	35	27	47
T11	0.025	0.031	0.040	0.044	25	0	33	47
T12	0.026	0.033	0.040	0.045	30	6	33	50
L8	0.913	0.977	0.993	0.997				
L12	0.955	0.986	0.987	0.993				
L16	0.928	0.970	0.995	0.998				
	After VBR/CBR							
VC	*eprcam*	*eprcam'*	*eprca*	*erica*	*eprcam*	*eprcam'*	*eprca*	*erica*
T1	0.032	0.175	0.173	0.173	10	0	7	0
T2	0.029	0.182	0.171	0.181	0	4	6	5
T3	0.030	0.181	0.161	0.177	3	3	0	2
T5	0.029	0.187	0.192	0.194	0	7	19	12
T6	0.031	0.188	0.190	0.208	7	7	18	20
T8	0.112	0.187	0.201	0.200	286	7	25	16
T9	0.111	0.185	0.207	0.198	283	6	29	14
T11	0.055	0.189	0.202	0.198	90	8	25	14
T12	0.053	0.190	0.204	0.206	83	9	27	19
L8	0.161	0.923	0.896	0.942				
L12	0.323	0.920	0.923	0.938				
L16	0.207	0.926	0.919	0.942				

Table 5.3: Throughput and Fairness Results Showing Impact of Measurement Interval (Case1)

	Throughput			Unfairness		
VC	1	1/10	1/100	1	1/10	1/100
T2	0.124	0.134	0.124	0	0	2
T3	0.122	0.134	0.122	0	0	0
T4	0.154	0.148	0.154	4	10	26
T5	0.153	0.146	0.153	5	9	25
T6	0.152	0.147	0.152	4	10	25
T7	0.150	0.144	0.150	4	7	23
T8	0.152	0.144	0.152	5	7	25
T9	0.143	0.142	0.143	4	6	17
T10	0.155	0.144	0.155	7	7	27
T11	0.150	0.145	0.150	6	8	23
T12	0.150	0.147	0.150	5	10	23
L8	0.945	0.966	0.962			
L12	0.936	0.954	0.949			
L16	0.944	0.961	0.958			

Table 5.4: Throughput and Fairness Results Showing Impact of Measurement Interval (Case2)

	Throughput			Unfairness		
VC	1	1/10	1/100	1	1/10	1/100
T1	.135	.134	.128	1	2	0
T2	.134	.134	.129	0	2	1
T3	.137	.132	.130	2	0	2
T5	.142	.146	.157	6	11	23
T6	.143	.147	.154	7	11	20
T8	.143	.150	.158	7	14	23
T9	.142	.146	.157	6	11	23
T11	.145	.147	.162	8	11	27
T12	.143	.150	.159	7	14	24
L8	.949	.951	.957			
L12	.948	.955	.959			
L16	.950	.955	.963			

Note: Each column heading is the length of measurement interval in milliseconds

Chapter 6

Concluding Remarks

6.1 Summary of This Dissertation

First of all, I presented a general overview of ATM technique specially the ATM layer functions for realizing broadband-ISDN (B-ISDN) in Chapter 1.

In Chapter 2 a comprehensive survey of most of the congestion control schemes proposed till to date is given and the issues related with the congestion control are identified. We find that the Enhanced Proportional Rate Control Algorithm (EPRCA) is a very interesting and promising algorithm in terms of performance achievement and implementability. However, due to typical handling of high congestion and no control in no congestion state, it gives rise to unfairness in bandwidth allocation to the contending sources.

In Chapter 3, it is stated that in EPRCA, the typical handling of high congestion state and then the subsequent no control in no congestion state resulted in substantial unfairness towards end-to-end traffic (Section 3.5). I propose some modifications in EPRCA to solve these problems and call the new algorithm as Modified Enhanced Proportional Rate Control Algorithm (EPRCAM) which is described in Section 3.6. A comprehensive performance comparison between EPRCA and EPRCAM is also given in this chapter.

In Chapter 4, I state that as the next step, Explicit Rate Indication for Congestion Avoidance (ERICA) algorithm is investigated which is being extensively discussed at present. I describe some problems in ERICA which are already identified by its original proposer and which give rise to unfairness in bandwidth utilization by the contending sources. However, it is discovered by us that the level of unfairness due to these problems is highly underestimated by the original proposer. I also find new problems in ERICA which produce unfairness (Section 4.4). Due to these problems and its high implementation complexity, I establish in this chapter that ERICA is outperformed by EPRCAM. I also give performance comparison of EPRCA, ERICA and EPRCAM in this chapter.

In Chapter 5, I explain that any congestion control algorithm for ABR service in ATM networks has to be robust so that when large amount of bandwidth is suddenly used or relinquished by the higher priority classes (for example VBR/CBR), its performance in terms of throughput and fairness must not be highly affected. I find that unfortunately the throughput and fairness performance of EPRCAM gets severely affected when a large amount of bandwidth is used or relinquished by the VBR/CBR connections. Further modifications are proposed in Section 5.4 to handle the above said problem of EPRCAM. It is shown in Section 5.6 that after proposed changes, the performance of EPRCAM is highly satisfactory against other congestion control algorithms.

6.2 Future Work

A lot of research in the field of congestion control in the ATM networks has done. The concepts developed for the congestion control may have even influenced the proposals of similar concepts of congestion control in the upper layers like TCP/IP.

Though, sufficient standardization has already been done by the International Standardization forums like the ATM Forum etc, a lot of standardization is yet to be done. Here, I state some of the hot topics of ATM networks on which research is going on these days.

1. Performance of upper layer protocols (e.g TCP/IP) over ATM networks: A lot of research has been done on this topic and many new avenues of research have come. Many problems are still unresolved in this field. For example, the double congestion control that is one in the ATM layer and the other in TCP/IP layer may result in unpredictable performance.

2. Guarantee of QoS in the ATM network: ATM networks are supposed possess service classes which provide guarantee of QoS. However, complex unresolved issues still exists. These issues must be resolved to realize implementation of guarantee of QoS in the ATM networks.

Bibliography

[1] A. J. McAuley, "Reliable Broadband Communication Using a Burst Erasure Correcting Code," *Proc. ACM SIGCOMM '90*, pp. 297–306, August 1990.

[2] H. Armbrïster and H. J. Rothamel, "Breitbandanwendungen unddienste - Qualitative and quantitative Anforderungen an künftige Netze," *Nachrichtentechnische Zeitschrift*, vol. 43, no. 3, pp. 150–159, March 1990.

[3] F. Bonomi and K. W. Fendick, "The Rate-Based Flow Control Framework for the Available Bit Rate ATM Service," *IEEE Network*, vol. 9, no. 2, pp. 25–39, 1995.

[4] R. Händel, M. N. Huber and S. schröder, *ATM Networks (Second edition): Concepts, Protocols, Applications*, Addition-Wesley publishing company, 1995.

[5] H. Heffes and D. M. Lucantoni, "A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Multiplexer Performance," *IEEE J.Selected Areas in Commun.*, vol. SAC-4, no. 6, pp.856–868, September 1986.

[6] ITU-T: COM XVIII-228-E, Geneva, March, 1984.

[7] ITU-T: Recommendation I.113. "Vocabulary of Terms for Broadband Aspects of ISDN," Rev. 1, Geneva, 1991.

[8] ITU-T: Recommendation I.121. "Broadband Aspects of ISDN," Rev. 1, Geneva, 1991.

[9] ITU-T: Recommendation I.321. "B-ISDN Protocol Reference Model and its Application," Geneva, 1993.

[10] ITU-T: Recommendation I.362. "B-ISDN ATM Adaptation Layer (AAL) Functional Description," Rev. 1, Geneva, 1993.

[11] ITU-T: Recommendation I.363. "B-ISDN ATM Adaptation Layer Specification," Rev. 1, Geneva, 1993.

[12] ITU-T: Recommendation I.371. "Traffic Control and Congestion Control in B-ISDN," Rev. 1, Geneva, 1993.

[13] ITU-T: Recommendation I.432. "B-ISDN User-Network Interface - Physical Layer Specification," Rev. 1, Geneva, 1993.

[14] J. Jaffe, "Bottleneck Flow Control," IEEE Trans. Comm., Vol. COM-29, No. 7, pp.954-962.

[15] R. Jain, "Myth About Congestion Management in High-Speed Networks," *Information Networks and Data Communication, IV, Elsevier Science Publishers B. V. (M. Tienari and D. Khakhar Editors)*, pp. 55–70, 1992.

[16] H. Kroner, "Comparative Performance Study of Space Priority Mechanisms for ATM Networks," *Proc. Infocom '90*, pp. 1136–1143, San Francisco, 1990.

[17] H. T. Kung and R. Morris, "Credit-Based Flow Control for ATM Networks," *IEEE Network*, vol. 9, no. 2, pp. 40–48, 1995.

[18] Martin De Prycker, *Asynchronous Transfer Mode (Third edition): Solution for Broadband ISDN*, Prentice Hall, 1995.

[19] K. Y. Siu and R. Jain, "A Brief Overview of ATM: Protocol Layers, LAN Emulation, and Traffic Management," *ACM SIGCOMM Computer Communication Review*, vol. 25, pp. 6–20, April 1995.

[20] C. Q. Yang and A. V. S. Reddy, "A Taxonomy for Congestion Control Algorithms in Packet Switching Networks," *IEEE Network*, vol. 9, no. 4, pp. 34–45, 1995.

[21] P. E. Boyer and D. P. Tranchier, "A Reservation Principle with Applications to the ATM Traffic Control," Computer Networks and ISDN Systems, Vol.24, 1992, pp.321-334.

[22] D. Kataria, "Comments on Rate-Based Proposal," AF-TM 94-0384, May 1994.

[23] R. Jain, "A Delay-Based Approach for Congestion Avoidance in Interconnected Hetrogenous Computer Networks," Computer Communications Review, Vol.19, No.5, October 1989, pp.56-71.

[24] P. Newman and G. Marshall, "BECN Congestion Control," AF-TM 94-789R1, July 1993.

[25] P. Newman and G. Marshall, "Update on BECN Congestion Control," AF-TM 94-855R1, September 1993.

[26] P. Newman, "Traffic Management for ATM Local Area Networks," IEEE Communications Magazine, August 1994, pp.44-50.

[27] K. K. Ramakrishnan, "Issues with backward Explicit Congestion Notification based Congestion Control," 93-870, September 1993.

[28] A. Romanov, "A Performance Enhancement for Packetized ABR and VBR+ Data," AF-TM 94-0295, March 1994.

[29] K. K. Ramakrishnan and J. Zavgren, "Preliminary Simulation Results of Hop-by-hop/VC Flow Control and Early Packet Discard," AF-TM 94-0231, March 1994.

[30] H. Tzeng and K. Siu, "A Class of Proportional Rate Control Schemes and Simulation Results," AF-TM 94-0888, September 1994.

[31] K. Siu and H. Tzeng, "Adaptive Proportional Rate Control for ABR Service in ATM Networks," University of California, Irvine, Technical Report 94-07-01.

[32] H. Tzeng and K. Siu, "Enhanced Credit-Based Congestion Notification (ECCN) Flow Control for ATM Network," AF-TM 94-0450, May 1994.

[33] B. Lyles and A. Lin, "Definition and Preliminary Simulation of a Rate-based Congestion Control Mechanism with Explicit Feedback of Bottleneck Rates," AF-TM 94-0708, July 1994.

[34] M. Hluchyj, et al., "Closed-Loop Rate-Based Traffic Management," AF-TM 94-0211R, April 1994.

[35] J. Bennet and G. Tom Des Jardins, "Comments on the July PRCA Rate Control Baseline," AF-TM 94-0682, July 1994.

[36] A. Charny, D. Clark, and R. Jain, "Congestion Control with Explicit Rate Indication," AF-TM 94-0692, July 1994, Updated version published in Proceedings ICC'95, June 1995.

[37] R. Jain, "Congestion Control in Computer Networks: Issues and Trends," IEEE Network Magazine, May 1990, pp. 24-30.

[38] R. Jain, S. Kalyanaraman, and R. Viswanathan, "The OSU Scheme for Congestion Avoidance Using Explicit Rate Indication," AF-TM 94-0883, September 1994.

[39] R. Jain, S. Kalyanaraman, and R. Viswanathan, "The EPRCA+ Scheme" AF-TM 94-0988, October 1994.

[40] R. Jain, S. Kalyanaraman, and R. Viswanathan, "The Transient Performance: EPRCA vs EPRCA++," AF-TM 94-1173, November 1994.

[41] R. Jain, S. Kalyanaraman, and R. Viswanathan, "A Sample Switch Algorithm," AF-TM 95-0178R1, February 1995.

[42] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, F. Lu, "ERICA+: Extensions to the ERICA Switch Algorithm," AF-TM 95-1145R1, October 1995.

[43] A. W. Barnhart, "Explicit Rate Performance Evaluation," AF-TM 94-0983R1, October 1994.

[44] M.A. Hashmani, Kenji Kawahara, Hideki Sunahara, and Yuji Oie, "ABR Congestion Control in ATM Networks and Proposal of EPRCAM," "Special Issue on Performance and Quality of Service (QoS) of Multimedia Networks" of IEICE, Japan, to be published in November 1998, available at http://minatow3.aist-nara.ac.jp/ manzoo-h/

[45] M.A. Hashmani, Kenji Kawahara, Hideki Sunahara, and Yuji Oie, "Issues of Congestion Control & Notification Schemes in ATM Networks and Proposal of EPRCAM," the proceedings of the IEEE International Conference on Communications (ICC) 1998, June 1998.

[46] M.A. Hashmani, Kenji Kawahara, Hideki Sunahara, and Yuji Oie, "Improved Fairness Performance over ABR Service in ATM Networks: Performance Comparison of ERICA and EPRCAM," submitted to International Journal of Communication Systems, U.K, available at http://minatow3.aist-nara.ac.jp/ manzoo-h/

[47] Sonia Fahmy, Raj Jain, Shivkumar Kalyanaraman, Rohit Goyal and Bobby Vandalore, "On Determining the Fair Bandwidth Share for ABR Connections in ATM Networks," To appear in proceedings of the IEEE International Conference on Communications (ICC) 1998, June 1998. Available at http://www.cis.ohio-state.edu/ jain/papers/neff.htm

[48] "ATM Traffic Management Specification Version 4.0," AF-TM 0056.0000, April 1996.

[49] H. Ohsaki, M. Murata, H. Suzuki, C. Ikeda, and H. Miyahara, "Rate-Based Congestion Control for ATM Networks," ACM SIGCOMM, Volume 25, Number 2 (April 1995).

[50] H.T. Kung, and A. Chapman, "Credit-based flow control for ATM networks: Credit update protocol, adaptive credit allocation, and statistical multiplexing," The proceedings of SIGCOMM '94, vol.24, pp.101-114, October 1994.

[51] D. Sisalem, and H. Schulzrinne, "End-to-End Rate Control in ABR," in First Workshop on ATM Traffic Management (WATM'95), (Paris, France), pp. 281-287, IFIP WG 6.2, Dec. 1995.

[52] Falvio Bonomi, and Kerry W. Fendick, "The Rate-Based Flow Control Framework for the Available Bit Rate ATM Service," IEEE Network, March/April 1995.

[53] Raj Jain, Sonia Fahmy, Shivkumar Kalyanaraman, and Rohit Goyal, "ABR Switch Algorithm Testing: A Case Study with ERICA," ATM Forum/96-1267, October 1996.

[54] R. Jain et al., "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks, Part II: Requirements and Performance Evaluation," submitted to IEEE/ACM Transactions on Networking, January 1997.

[55] R. Jain et al., "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks, Part I: Description," submitted to IEEE/ACM Transactions on Networking, January 1997.

[56] Shivkumar Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, and Bobby Vandalore, "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks," submitted to IEEE/ACM Transactions on Networking, November 1997.

[57] Raj Jain, "ERICA Switch Algorithm: A Complete Description," ATM Forum/96-1172.

[58] Raj Jain, "Congestion Control and Traffic Management in ATM Networks: Recent Advances and a Survey," January 1995.

[59] L. Roberts, "Enhanced PRCA (proportional rate-control algorithm)," AF-TM 94-0735R1, August 1994.

[60] L. Roberts, "The Benefits of Rate-Based Flow Control for ABR Service," AF-TM 94-0796, September 1994.

[61] L. Roberts et al., "New Pseudocode for Explicit Rate Plus EFCI Support," AF-TM 94-0974, October 1994.

[62] L. Roberts, "Rate-based Algorithm for Point to Multipoint ABR Service," AF-TM 94-0772R1, November 1994.