CONGESTION CONTROL FOR THE AVAILABLE BIT RATE (ABR) SERVICE IN ASYCHRONOUS TRANSFER MODE (ATM) NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

HÜLYA BOZKURT ÖZZAMAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

APRIL 2005

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr.Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. İsmet ERKMEN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof. Dr. Kemal LEBLEBİCİOĞLU
Supervisor

Examining Committee Members

Prof. Dr. Uğur HALICI	(METU, EEE)	
Prof. Dr. Kemal LEBLEBİCİOĞLU	(METU, EEE)	
Asst. Prof Dr. Cüneyt F.BAZLAMAÇCI	(METU, EEE)	
Dr. Şenan Ece SCHMIDT	(METU, EEE)	
Dr. Can Erkin ACAR	(PRO-G)	

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Hülya Bozkurt Özzaman

Signature:

ABSTRACT

CONGESTION CONTROL FOR THE AVAILABLE BIT RATE (ABR) SERVICE IN ASYCHRONOUS TRANSFER MODE (ATM) NETWORKS

Bozkurt Özzaman, Hülya
M.S., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Kemal Leblebicioğlu

April 2005, 94 pages

Congestion control is concerned with allocating the resources in a network such that the network can operate at an acceptable performance level when the demand exceeds or is near the capacity of the network resources. These resources include bandwidths of links, buffer space (memory) and processing capacity at intermediate nodes. Although resource allocation is necessary even at low load, the problem becomes more important as the load increases. Without proper congestion control mechanisms, the throughput may be reduced considerably under heavy load.

Future applications are expected to require increasingly higher bandwidth and generate a heterogeneous mix of network traffic. ATM network is potentially capable of supporting all classes of traffic (e.g., voice, video, and data) and have multiple service classes allow audio, video and data to share the same network. Of these, the Available Bit Rate (ABR) service class is designed to efficiently support data traffic.

Switch algorithms have been the most investigated topic of ABR. This has happened because the specification of ABR given by the ATM Forum allows a diversity of switch algorithms to be implemented. These range from the simplest binary switches to the more complex ER switches.

The major part of this thesis has been devoted to ABR. First an introduction to the concept of congestion control and a brief literature survey of congestion control for ABR service of ATM networks are presented. Then two proposed congestion control mechanisms for the ABR service class in ATM networks are examined by means of simulation, showing the different degree of performance and complexity.

The simulation results presented in this thesis were obtained using a network simulator written in C++. This network simulator is a small event driven program. Analytical results were derived for different network configurations and different scenarios using this program.

Keywords: Congestion Control, ATM, ABR Service Category, Simulation

ÖZ

ATM AĞLARDA KULLANILABİLİR BİT MİKRARI SERVİSİ İÇİN TIKANIKLIK KONTROLÜ

Bozkurt Özzaman, Hülya
Yüksek Lisans, Elektrik Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Kemal Leblebicioğlu

Nisan 2005, 94 sayfa

Tıkanıklık kontrolü, ağ kaynaklarına olan talebin kapasiteye yakın ya da üzerinde olması durumunda ağın kabul edilebilir bir performans seviyesinde çalışabilmesi için, kaynakların tahsis edilmesi konusunu inceler. Bu kaynaklar hattın band genişliğini, tampon bellek bölgesini ve ara düğümlerdeki işleme kapasitesini içerir. Kaynak tahsisi az yüklerde de gerekli olmasına rağmen, yük arttıkça problem daha önemli bir hal alır. Doğru tıkanıklık kontrol mekanizması kullanılmadığı takdirde, fazla yükde ağ performansı düşebilir.

Gelecekteki uygulamaların yüksek band genişliğini ve her çeşit ağ trafiğini desteklemesi beklenmektedir. ATM ağları her çeşit trafiği desteklemektedir ve ATM ağlarının ses görüntü ve veri bilgisinin aynı ağı paylaşmasına olanak tanıyan farklı servis çeşitleri vardır. Bunlardan, kullanılabilir bit miktarı (Available Bit Rate) servis tipi veri trafiğini verimli bir şekilde desteklemesi için tasarlanmıştır.

Anahtar algoritmaları ABR'nin en çok araştırma yapılan konusudur. ATM Forum tarafından ABR için belirlenen özellikler çok çeşitli anahtar algoritmalarının geliştirilmesine neden olmuştur. Bu algoritmalar basit ikili anahtarlardan ER anahtarlara kadar olan çeşitleri kapsar.

Bu tezin büyük bölümünde ABR'ye yer verilmiştir. Önce tıkanıklık kontrolü kavramına bir giriş yapılmış ve ATM ağlarda ABR servis tipi için özet literatür taraması yapılmıştır. Daha sonra ATM ağlarda ABR servis tipi için öne sürülmüş iki tıkanıklık kontrolü algoritması farklı derecelerdeki performans ve karışıklıklarını gösteren simülasyonlarla incelenmiştir.

Bu tezde sunulan simülasyon sonuçları C++ ile yazılan bir ağ simülatörü ile elde edilmiştir. Bu ağ simülatörü olay tetikleyen bir programdır. Bu program kullanılarak farklı ağ biçimleri ve farklı senaryolar için analitik sonuçlar çıkarılmıştır.

Anahtar Kelimeler: Tıkanıklık Kontrolü, ATM, ABR Servis Kategorisi,
Simülasyon

To My Family

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude to my supervisor, Prof. Dr. Kemal Leblebicioğlu, for his guidance, advice, criticism, encouragement and insight throughout the research.

I could never achieve this goal without the support of my husband and my family. I thank them for their love, encouragement and full support.

Thanks are due to my good friends Güray, Emre, Nevra, Apo, Somer, Kasım and Koray who helped, encouraged and understood me during this thesis work.

I also would like to thank HAVELSAN Sim.Sist.Dir. for giving me the opportunity to pursue my M.S. in METU.

Finally, I would like to thank again, my father, my mother and also my husband, Gökhan, for their love and patience. Without them I might never have finished writing this thesis.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ABR	Available Bit Rate
ACR	Allowed Cell Rate
ADTF	ACR Decrease Time Factor
ATM	Asynchronous Transfer Mode
B-ISDN	Broadband ISDN
CAC	Connection Admission Control
CBR	Constant Bit Rate
CCR	Current Cell Rate
CDV	Cell Delay Variation
CDVT	Cell Delay Variation Tolerance
CLR	Cell Loss Ratio
CLP	Cell Loss Priority
CTD	Cell Transfer Delay
EFCI	Explicit Forward Congestion Indication
ER	Explicit Rate
ERICA	Explicit Rate Indication for Congestion Avoidance
ICR	Initial Cell Rate
ISO	International Organization for Standardization
MCR	Minimum Cell Rate
NI	No Increase
nrt-VBR	Non-real time Variable Bit Rate
PCR	Peak Cell Rate
QoS	Quality of Service
RDF	Rate Decrease Factor
RIF	Rate Increase Factor
RM	Resource Management

rt-VBR	Realtime Variable Bit Rate
RTT	Round Trip Time
SCR	Sustainable Cell Rate
SES	Source End System
TCR	Target Cell Rate
UBR	Unspecified Bit Rate
UPC	Usage Parameter Control
VBR	Variable Bit Rate
VC	Virtual Circuit
VP	Virtual Path

CHAPTER 1

INTRODUCTION

Congestion control is concerned with allocating the resources in a network such that the network can operate at an acceptable performance level when the demand exceeds or is near the capacity of the network resources. These resources include bandwidths of links, buffer space (memory) and processing capacity at intermediate nodes. Although resource allocation is necessary even at low load, the problem becomes more important as the load increases. Without proper congestion control mechanisms, the throughput may be reduced considerably under heavy load.

The major part of this thesis has been devoted to congestion control for ABR service of ATM networks. The thesis is organized as follows: Chapter 1 presents an introduction to congestion, congestion control, and traffic management in ATM networks. A brief introduction to ATM service categories is also presented. Chapter 2 discussed ABR traffic management framework. Behaviors of source, switch, and destination in ABR service are explained. Chapter 3 introduces a well-known algorithm called ERICA. The ERICA algorithm is concerned with the fair and efficient allocation of the available bandwidth to all contending sources. In chapter 4, to compute data rates for ABR sources we introduced a control-based mathematical model that helps us to implement an explicit-rate congestion control algorithm. In chapter 5 simulation results are analyzed and a comparison between the ERICA algorithm and the control approach is given. Finally, chapter 6 concludes the thesis.

1.1 What is Congestion?

The system is said to be congested if it is being offered more traffic than its rated capacity. The term “congestion control” is generally used to refer to the action of regulating various flows within a network. This control action arises due to the scarcity of the network resources, such as link capacities. Congestion occurs when the demand is greater than the available resources. Therefore, it is believed that as resources become less expensive, the problem of congestion will be solved automatically. This has led to the following myths [1]:

- Congestion is caused by a shortage of buffer space and will be solved when memory becomes cheap enough to allow infinitely large memories.
- Congestion is caused by slow links. The problem will be solved when high-speed links become available.
- Slow processors cause congestion. The problem will be solved when the speed of the processor is improved.
- If not one, then all of the above developments will cause the congestion problem to go away.

Contrary to these beliefs, without proper protocol redesign, the above developments may lead to more congestion and thus reduce performance because congestion is a dynamic problem and cannot be solved with static solutions alone. We need protocol designs that protect networks in the event of congestion.

1.2 Congestion Control in ATM Networks

In ATM networks, the information is transmitted using short fixed-length cells. This network is suitable for integrated traffic, which includes voice, video and data. If proper traffic management is done, ATM can operate at an efficient level to meet different quality of service (QoS) desired by different types of traffic.

The specification of ATM networks is given by the ATM Forum. The ATM Forum is an international non-profit organization formed with the objective of accelerating the use of ATM (Asynchronous Transfer Mode) products and services through a rapid convergence of interoperability specifications. In addition, the Forum promotes industry cooperation and awareness. Members of ATM Forum consist of network equipment providers, service providers, software developers and customers [23].

1.2.1 Basic Principles of ATM

ATM network is potentially capable of supporting all classes of traffic (e.g., voice, video, and data) and it uses fixed-length cells to transmit information. The cell consists of 48 bytes of payload and 5 bytes of header [3].

ATM network is connection-oriented. It sets up virtual channel connection (VCC) before transmitting information.

ATM resources such as bandwidth and buffers are shared among users; they are allocated to the user only when they have something to transmit. So the network uses statistical multiplexing to improve the effective throughput. Providing desired QoS for different applications is very complex. For example, voice is delay-sensitive but not loss-sensitive, data is loss-sensitive but not delay-sensitive, while some other applications may be both delay-sensitive and loss-sensitive.

To make it easier to manage, the traffic in ATM is divided into five service classes. These service classes are defined in the following [5].

1.2.1.1 Constant Bit Rate (CBR) Service Category Definition

The Constant Bit Rate service category is used by connections that request a static amount of bandwidth that is continuously available during the connection lifetime. This amount of bandwidth is characterized by a Peak Cell Rate (PCR) value.

In the CBR capability, the source can emit cells at the Peak Cell Rate at any time and for any duration and the QoS commitments still pertain. CBR service is intended to support real-time applications requiring tightly constrained delay variation (e.g., voice, video, circuit emulation) but is not restricted to these applications. In the CBR capability, the source may emit cells at, or below the negotiated Peak Cell Rate (and may also even be silent), for periods of time.

1.2.1.2 Real-Time Variable Bit Rate (rt-VBR) Service Category Definition

The real-time VBR service category is intended for real-time applications, i.e., those requiring tightly constrained delay and delay variation, as would be appropriate for voice and video applications. rt-VBR connections are characterized in terms of a Peak Cell Rate (PCR), Sustainable Cell Rate (SCR), and Maximum Burst Size (MBS). Sources are expected to transmit at a rate, which varies with time.

1.2.1.3 Non-Real-Time (nrt-VBR) Service Category Definition

The non-real-time VBR service category is intended for non-real-time applications which have bursty traffic characteristics and which are characterized in terms of a PCR, SCR, and MCR. For those cells, which are transferred within the traffic contract, the application expects a low cell loss ratio. No delay bounds are associated with this service category.

1.2.1.4 Unspecified Bit Rate (UBR) Service Category Definition

The Unspecified Bit Rate (UBR) service category is intended for non-real-time applications, i.e., those not requiring tightly constrained delay and delay variation. Examples of such applications are traditional computer communications applications, such as file transfer and email.

1.2.1.5 Available Bit Rate (ABR) Service Category Definition

ABR is an ATM layer service category for which the limiting ATM layer transfer characteristics provided by the network may change subsequent to connection establishment. A flow control mechanism is specified which supports several types of feedback to control the source rate in response to changing ATM layer transfer characteristics. This feedback is conveyed to the source through specific control cells called Resource Management Cells, or RM-cells. It is expected that an end-system that adapts its traffic in accordance with the feedback will experience a low cell loss ratio and obtain a fair share of the available bandwidth according to a network specific allocation policy. The ABR service does not require bounding the delay or the delay variation experienced by a given connection. ABR service is not intended to support real-time applications.

On the establishment of an ABR connection, the end-system shall specify to the network both a maximum required bandwidth and a minimum usable bandwidth. These shall be designated as peak cell rate (PCR), and the minimum cell rate (MCR), respectively. The MCR may be specified as zero. The bandwidth available from the network may vary, but shall not become less than MCR.

1.2.2 Traffic Management and Congestion Control

A key issue in ATM and in any network architecture design is resource management, i.e., how to make the best use of available resources.

Traffic management is a resource management problem, which deals exclusively with the mechanisms required to control traffic on the network [4]. A related problem is "congestion" which occurs when the aggregate demand for a resource (typically link bandwidth) exceeds the available capacity of the resource. In other words, congestion happens whenever the demand is more than the available capacity:

Demand > Available Capacity

Congestion management involves the design of mechanisms and schemes to statically limit the demand-capacity mismatch, or dynamically control traffic sources when such a mismatch occurs. Congestion is a problem associated with the dynamics of the network load and capacity, static solutions such as allocating more buffers, or providing faster links, or faster processors does not solve the problem.

Observe that congestion management deals with the problem of matching the demand and capacity for a single network traffic class. Traffic management, even for a single traffic class, deals with the problem of ensuring that the network bandwidth, buffer and computational resources are efficiently utilized while meeting the various Quality of Service (QoS) guarantees given to sources as part of a traffic contract. The general problem of network traffic management involves all the available traffic classes. In ATM networks, the general traffic management problem involves the mechanisms needed to control the multiple classes of traffic (like CBR, VBR, ABR and UBR) while ensuring that all the traffic contracts are met. The components of traffic management other than congestion management schemes include scheduling mechanisms, traffic contract negotiation, admission control, and traffic policing. In this thesis, we address the problem of designing traffic management mechanisms for one class - the ABR service class in ATM networks.

1.2.3 Connection Parameters

1.2.3.1 Quality of Service Parameters

A set of parameters is negotiated when a connection is set up on ATM networks [3]. These parameters are used to measure the Quality of Service (QoS) of a connection. The network should guarantee the QoS by meet certain values of these parameters.

- Cell Transfer Delay (CTD): The delay experienced by a cell between the first bit of the cell is transmitted by the source and the last bit of the cell is received by the destination. Maximum Cell Transfer Delay (Max CTD) and Mean Cell Transfer Delay (Mean CTD) are used.
- Peak-to-peak Cell Delay Variation (CDV): The difference of the maximum and minimum CTD experienced during the connection.
- Cell Loss Ratio (CLR): The percentage of cells that are lost in the network due to error or congestion and are not received by the destination.

1.2.3.2 Usage Parameters

Another set of parameters is also negotiated when a connection is set up. These parameters discipline the behavior of the user.

- Peak Cell Rate (PCR): The maximum instantaneous rate at which the user will transmit.
- Sustained Cell Rate (SCR): The average rate as measured over a long interval.
- Burst Tolerance (BT): The maximum burst size that can be sent at the peak rate.
- Maximum Burst Size (MBS): The maximum number of back-to-back cells that can be sent at the peak cell rate. BT and MBS are related as follows: Burst Tolerance = (MBS - 1)(1/SCR - 1/PCR)
- Minimum Cell Rate (MCR): The minimum cell rate desired by a user.

The QoS requirement for each class is different. The traffic management policy for them is different, too.

Among the service classes, ABR is commonly used for data transmissions, which requires low probability of loss and error. Due to the burstiness, unpredictability and huge amount of the data traffic, congestion control of this class is the most needed and is also the most studied.

ATM Forum Technical Committee specified the feedback mechanism for ABR flow control.

1.2.4 Generic Functions

Congestion control lies at the heart of the general problem of traffic management for ATM networks; in general, congestion arises when the incoming traffic to a specific link is more than the outgoing link capacity. The primary function of congestion control is to ensure good throughput and delay performance while maintaining a fair allocation of network resources to the users.

One way to classify congestion control schemes is based on the layer of the ISO/OSI reference model at which the scheme operates. For example, there are data link, routing, and transfer layer congestion control schemes. Typically, a combination of such schemes is used. The selection depends upon the severity and duration of congestion. Figure 1 shows how the duration of congestion affects the choice of the method. We can also say that the congestion control schemes can be classified by the time scale they operate upon: network design, connection admission control (CAC), routing, traffic shaping, end-to-end feedback control, hop-by-hop feedback control, buffering. The different schemes are functions on different severity of congestion as well as different duration of congestion. Figure 1 shows the congestion techniques for various congestion durations.

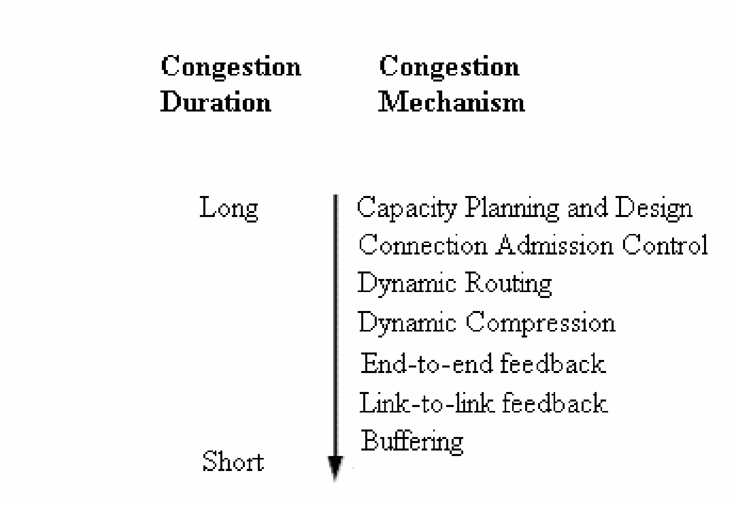

Figure 1 Congestion techniques for various congestion durations

No matter what kind of scheme is used, the following outstanding problems are the main difficulties that need to be treated carefully: the burstiness of the data traffic, the unpredictability of the resource demand and the large propagation delay versus the large bandwidth.

To meet the objectives of traffic control and congestion control in ATM networks, the ATM Forum Technical Committee suggests the following functions and procedures.

1.2.4.1 Connection Admission Control

Connection Admission Control (CAC) is defined as the set of actions taken by the network during the call set-up phase in order to determine whether a connection request can be accepted or should be rejected.

Based on the CAC algorithm, a connection request is progressed only when sufficient resources such as bandwidth and buffer space are available along the path of a connection. The decision is made based on the service category, QoS desired and the state of the network which means that the number and conditions of existing connections.

Routing and resource allocation are part of CAC when a call is accepted.

1.2.4.2 Usage Parameter Control

Usage Parameter Control (UPC) is defined as the set of actions taken by the network to monitor and control traffic at the end-system access. Its main purpose is to protect network resources from user misbehavior, which can affect the QoS of other connections, by detecting violations of negotiated parameters and taking appropriate actions.

1.2.4.3 Priority Control

The end-system may generate traffic flows of different priority using the Cell Loss Priority (CLP) bit. The network may selectively discard cells with low priority if necessary such as in congestion to protect, as far as possible, the network performance for cells with high priority.

1.2.4.4 Traffic Shaping

Traffic shaping is a mechanism that alters the traffic characteristics of a stream of cells on a connection to achieve better network efficiency while meeting the QoS objectives, or to ensure conformance at a subsequent interface. Examples of traffic shaping are peak cell rate reduction, burst length limiting, and reduction of CDV by suitably spacing cells in time, and queue service schemes. Traffic shaping may be performed in conjunction with suitable UPC functions.

1.2.4.5 Network Resource Management

Network Resource Management (NRM) is responsible for the allocation of network resources in order to separate traffic flows according to different service characteristics, to maintain network performance and to optimize resource utilization. This function is mainly concerned with the management of virtual paths in order to meet QoS requirements.

1.2.4.6 Frame Discard

If a congested network needs to discard cells, it may be better to drop all cells of one frame than to randomly drop cells belonging to different frames, because one cell loss may cause the retransmission of the whole frame, which may cause more traffic when congestion already happened. Thus, frame discard may help avoid congestion collapse and can increase throughput. If done selectively, frame discard may also improve fairness.

1.2.4.7 ABR Flow Control

As we have discussed before, the ABR service category uses the link capacity that is left over and is applied to transmit critical data that is sensitive to cell loss. That makes traffic management for this class the most challenging

The ATM Forum Technical Committee Traffic Management Working Group has worked hard on this topic, and next chapter gives some of the main issues and the current progress of this area.

CHAPTER 2

ABR CONGESTION CONTROL IN ATM

With the merger of telecommunication, entertainment and computer industries, computer networking has adopted a technology called Asynchronous Transfer Mode (ATM) networking. ATM networks have multiple service classes allow audio, video and data to share the same network. Of these, the Available Bit Rate (ABR) service class is designed to efficiently support data traffic.

In the Available Bit Rate Service Category the sources adapt their transmission rates to changing network conditions. Information about the state of the network, such as bandwidth availability, state of congestion, and impending congestion, is conveyed to the source through special control cells called RM (Resource Management) cells. An international non-profit organization, the ATM Forum has specified the way the feedback information is conveyed to the ABR sources. The standard specifies the source and destination behavior and several fields of RM-cells that can be used by the switches to control source rates. The exact algorithm used by the switches to make use of these fields is implementation specific.

Switches using the EFCI (Explicit Forward Congestion Indication) marking or relative rate marking mechanisms are referred to as binary switches. This first generation of ABR switches would adjust the transmission rate of the source by this simple binary indication [6].

Second generation of switches would perform a fairness algorithm to compute the transmission rate of each source and use Explicit Rate (ER) field of the RM-cell to convey this rate to the sources. These are called ER switches. ER switch algorithms have been studied in many contributions. Some of the proposed algorithms are based on control theory. Other switch algorithms are directly derived from the fair bandwidth allocation criteria.

2.1 ABR Rate Based Framework

ABR mechanisms allow the network to divide the available bandwidth fairly and efficiently among the active traffic sources [2]. In the ABR traffic management framework, the source end systems limit their data transmission to rates allowed by the network. The network consists of switches, which use their current load information to calculate the allowable rates for the sources. These rates are sent to the sources as feedback via resource management (RM) cells. RM cells are generated by the sources and travel along the data path to the destination end systems. The destinations simply return the RM cells to the sources. The components of the ABR traffic management framework are shown in Figure 2. In this chapter, we explain the source and destination end-system behaviors and their implications on ABR traffic management [7].

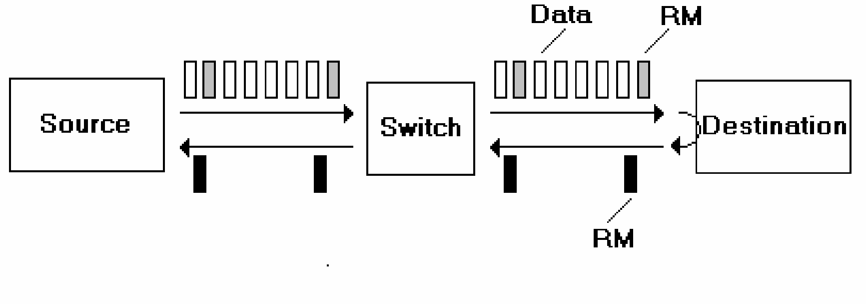

Figure 2 ABR traffic management model

The ABR traffic management model is called a "rate-based end-to-end closed-loop" model. The model is called "rate-based" because the sources send data at a specified rate. This is different from current packet networks (for example, TCP), where the control is "window based" and the sources limit their transmission to a particular number of packets. The ABR model is called "closed-loop" because there is a continuous feedback of control information between the network and the source. If more sources become active, the rate allocated to each source is reduced. The model used for CBR and VBR traffic, on the other hand, is "open-loop" in the sense that rates are negotiated at the beginning of the connection and do not change dynamically. Finally, the model is called "end-to-end" because the control cells travel from the source to the destination and back to the source [5].

2.1.1 Feedback Methods

There are three ways for switches to give feedback to the sources [2]:

In the first way, each cell header contains a bit called Explicit Forward Congestion Indication (EFCI), which can be set by a congested switch. Such switches are called "binary" or "EFCI" switches.

In the second way, RM cells have two bits in their payload, called the Congestion Indication (CI) bit and the No Increase (NI) bit, that can be set by congested switches. Switches that use only this mechanism are called "relative rate marking" switches.

In the third way, the RM cells also have another field in their payload called explicit rate (ER) that can be reduced by congested switches to any desired value. Such switches are called "explicit rate" switches.

Explicit rate switches normally wait for the arrival of an RM cell to give feedback to a source. However, under extreme congestion, they are allowed to generate an RM cell and send it immediately to the source. This optional mechanism is called "Backward Explicit Congestion Notification" (BECN).

2.2 ABR Parameters

At the time of connection setup, ABR sources negotiate several operating parameters with the network. The first among these is the peak cell rate (PCR). This is the maximum rate at which the source will be allowed to transmit on this virtual circuit (VC). The source can also request a minimum cell rate (MCR), which is the guaranteed minimum rate. The network has to reserve this bandwidth for the VC. During the data transmission stage, the rate at which a source is allowed to send at any particular instant is called the allowed cell rate (ACR). The ACR is dynamically changed between MCR and PCR. At the beginning of the connection, and after long idle intervals, ACR is set to initial cell rate (ICR). During the development of the RM specification, all numerical values in the specification were replaced by mnemonics. For example, instead of saying, "every 32nd cell should be an RM cell", the specification states, "Every Nrmth cell should be an RM cell." Here, Nrm is a parameter whose default value is 32. Some of the parameters are fixed while others are negotiated. A complete list of parameters used in the ABR mechanism is presented in Table 1.

Table 1 List of ABR parameters

Label	Expansion	Default Value
PCR	Peak Cell Rate	-
MCR	Minimum Cell Rate	0
ACR	Allowed Cell Rate	-
ICR	Initial Cell Rate	PCR
TCR	Tagged Cell Rate	10 cells/sec
Nrm	Number of cells between FRM cells	32
Mrm	Controls bandwidth allocation between FRM, BRM and Data cells	2
Trm	Upper Bound on Inter-FRM Time	100 ms
RIF	Rate Increase Factor	1/16
RDF	Rate Decrease Factor	1/16
ADTF	ACR Decrease Time Factor	500 ms
TBE	Transient Buffer Exposure	16777215
CRM	Missing RM-cell Count	TBE/Nrm
CDF	Cutoff Decrease Factor	1/16
FRTT	Fixed Round-Trip Time	-

2.3 Resource Management Cells

2.3.1 In-Rate and Out-of-Rate RM Cells

Most resource management cells generated by the sources are counted as part of their network load in the sense that the total rate of data and RM cells should not exceed the ACR of the source. Such RM cells are called "in-rate" RM cells. Under exceptional circumstances, switches, destinations, or even sources can generate extra RM cells. These "out-of-rate" RM cells are not counted in the ACR of the source and are distinguished by having their cell loss priority (CLP) bit set, which means that the network will carry them only if there is plenty of bandwidth and can discard them if congested. The out-of-rate RM cells generated by the source and switch are limited to 10 RM cells per second per VC. One use of out-of-rate RM cells is for BECN from the switches. Another use is for a source, whose ACR has been set to zero by the network, to periodically sense the state of the network. Out-of-rate RM cells are also used by destinations of VCs whose reverse direction ACR is either zero or not sufficient to return all RM cells received in the forward direction.

2.3.2 Forward and Backward RM cells

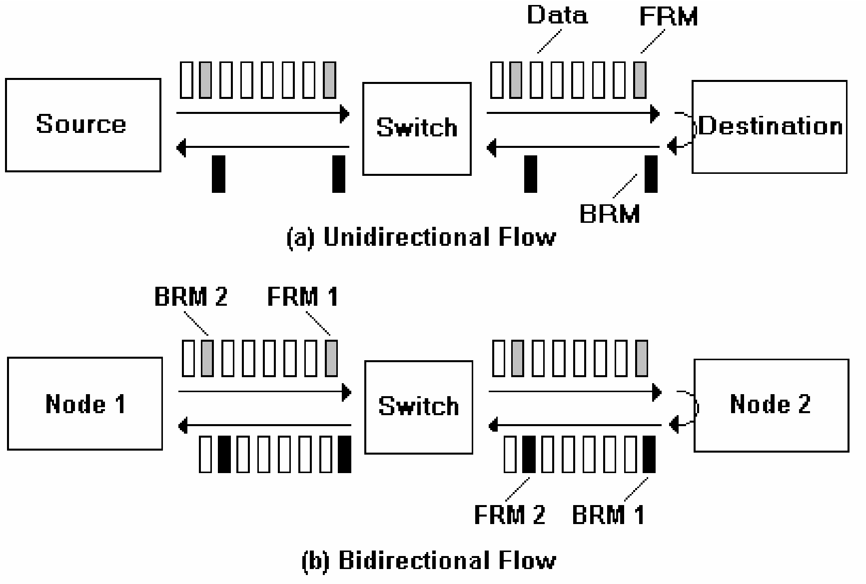

Figure 3 Forward and backward resource management cells (FRMs and BRMs)

Resource Management cells traveling from the source to the destination are called Forward RM (FRM) cells. The destination turns around these RM cells and sends them back to the source on the same VC. Such RM cells traveling from the destination to the source are called Backward RM (BRM) cells. Forward and backward RM cells are illustrated in Figure 3. Note that when there is bidirectional traffic, there are FRMs and BRMs in both directions on the Virtual Channel (VC). A bit in the RM cell payload indicates whether it is an FRM or BRM. This direction bit (DIR) is changed from 0 to 1 by the destination.

2.4 RM Cell Format

The complete format of the RM cells is shown in Table 2. Every RM cell has the regular ATM header of five bytes. The payload type indicator (PTI) field indicates that the cell is an RM cell. The protocol ID field, which is one byte long, is set to one for ABR connections. The direction (DIR) bit distinguishes forward and backward RM cells. The backward notification (BN) bit is set only in switch generated BECN cells. The congestion indication (CI) bit is used by relative rate marking switches. It may also be used by explicit rate switches under extreme congestion. The no increase (NI) bit is another bit available to explicit rate switches to indicate moderate congestion.

Table 2 Resource management (RM) cell fields

ATM Header	5 Bytes
Protocol ID	1 Byte
Direction	1bit
Backward Notification	1bit
Congestion Indication	1bit
No Increase	1bit
Request / Acknowledge	1bit
Reserved	3 bits
Explicit Rate	2 Bytes
Current Cell Rate	2 Bytes
Minimum Cell Rate	2 Bytes
Queue Length	4 Bytes
Sequence Number	4 Bytes
Reserved	30.75
CRC-10	10 bits

The Current Cell Rate (CCR) field is used by the source to indicate to the network its current rate. Some switches may use the CCR field to determine a VC's next allocation while others may measure the VC's rate and not trust CCR. The minimum cell rate (MCR) field is redundant in the sense that like PCR, ICR, and other parameters it does not change during the life of a connection.

The Explicit Rate (ER) field, the CI and the NI fields are used by the network to give feedback to the sources. The ER field indicates the maximum rate allowed to the source. When there are multiple switches along the path, the feedback given by the most congested link is the one that reaches the source. Data cells also have an Explicit Forward Congestion Indication (EFCI) bit in their headers, which may be set by the network when it experiences congestion. The destination saves the EFCI state of every data cell. If the EFCI state is set when it turns around an RM cell, it uses the CI bit to give (a single bit) feedback to the source. When the source receives the RM cell from the network, it adjusts its ACR using the ER, CI, NI values, and source parameters.

2.5 Source End System Rules

TM4.0 [5] specifies 13 rules that the sources have to follow. The following items define the source behavior for CLP=0 and CLP= 1 cell streams of a connection. By convention, the CLP=0 stream is referred to as in-rate, and the CLP=1 stream is referred to as out-of-rate. Data cells shall not be sent with CLP=1.

Source Rule 1: Sources should always transmit at a rate equal to or below their computed ACR. The ACR cannot exceed PCR and need not go below MCR. Mathematically,

$$MCR \le ACR \le PCR \tag{2.1}$$

$$SourceRate \le ACR \tag{2.2}$$

Source Rule 2: At the beginning of a connection, sources start at ICR. The first cell is always an in-rate forward RM cell. This ensures that the network feedback will be received as soon as possible.

Source Rule 3: After the first in-rate forward RM-cell, in-rate cells shall be sent in the following order:

a) The next in-rate cell shall be a forward RM-cell if and only if, since the last in-rate forward RM-cell was sent, either:

i) at least Mrm in-rate cells have been sent and at least Trm time has elapsed, or ii) Nrm-1 in-rate cells have been sent.

b) The next in-rate cell shall be a backward RM-cell if condition (a) above is not met, if a backward RM-cell is waiting for transmission, and if either:

i) no in-rate backward RM-cell has been sent since the last in-rate forward RM-cell, or ii) no data cell is waiting for transmission.

c) The next in-rate cell sent shall be a data cell if neither condition (a) nor condition (b) above is met, and if a data cell is waiting for transmission.

Source Rule 4: Cells sent in accordance with source behaviors #1, #2, and #3 shall have CLP=0.

Source Rule 5: Before sending a forward in-rate RM-cell, if ACR > ICR and the time T that has elapsed since the last in-rate forward RM-cell was sent is greater than ADTF, then ACR shall be reduced to ICR.

Source Rule 6: Before sending an in-rate forward RM-cell, and after following behavior #5 above, if at least CRM in-rate forward RM-cells have been sent since the last backward RM-cell with BN=0 was received, then ACR shall be reduced by at least ACR*CDF, unless that reduction would result in a rate below MCR, in which case ACR shall be set to MCR.

Source Rule 7: After following behaviors #5 and #6 above, the ACR value shall be placed in the CCR field of the outgoing forward RM-cell, but only in-rate cells sent after the outgoing forward RM-cell need to follow the new rate.

Source Rules 8: When a backward RM-cell (in-rate or out-of-rate) is received with CI=1, then ACR shall be reduced by at least ACR*RDF, unless that reduction would result in a rate below MCR, in which case ACR shall be set to MCR. If the backward RM-cell has both CI=0 and NI=0, then the ACR may be increased by no more than RIF*PCR, to a rate not greater than PCR. If the backward RM-cell has NI=1, the ACR shall not be increased.

Source Rule 9: When a backward RM-cell (in-rate or out-of-rate) is received, and after ACR is adjusted according to source behavior #8, ACR is set to at most the minimum of ACR as computed in source behavior #8, and the ER field, but no lower than MCR.

Source Rule 10: When generating a forward RM-cell, the source shall assign values to the various RM-cell fields.

Source Rule 11: Forward RM-cells may be sent out-of-rate (i.e., not conforming to the current ACR). Out-of-rate forward RM-cells shall not be sent at a rate greater than TCR.

Source Rule 12: A source shall reset EFCI on every data cell it sends.

Source Rule 13: The source may implement a use-it-or-lose it policy to reduce its ACR to a value which approximates the actual cell transmission rate.

2.6 Destination End System Rules

Destination Rule 1: Destinations should monitor the EFCI bits on the incoming cells and store the value last seen on a data cell.

Destination Rule 2: Destinations are required to turn around the forward RM cells with minimal modifications as follows: the DIR bit is set to "backward" to indicate that the cell is a backward RM-cell; the BN bit is set to zero to indicate that the cell was not generated by a switch; the CCR and MCR field should not be changed. If the last cell has EFCI bit set, the CI bit in the next BRM is set and the stored EFCI state is cleared.

If the destination has internal congestion, it may reduce the ER or set the CI or NI bits just like a switch.

Destination Rules 3-4: The destination should turn around the RM cells as fast as possible. However, an RM cell may be delayed if the reverse ACR is low. In such cases destination rules 3 and 4 specify that old out-of-date information can be discarded. The destinations are allowed a number of options to do this.

If the reverse direction ACR is non-zero, then a backward RM cell will be scheduled for in-rate transmission. Transmitting backward RM cells out-of-rate ensures that the feedback is sent regularly even if the reverse ACR is low or zero (for example, in unidirectional VCs). Note that there is no specified limit on the rate of such "turned around" out-of-rate RM cells. However, the CLP bit is set to 1 in the out-of-rate cells, which allows them to be selectively dropped by the switch if congestion is experienced.

Destination Rule 5: Sometimes a destination may be too congested and may want the source to reduce its rate immediately without having to wait for the next RM cell. Therefore, like a switch, the destinations are allowed to generate BECN RM cells. Also, as in the case of switch generated BECNs, these cells may not ask a source to increase its rate (CI bit is set). These BECN cells are limited to 10 cells/s and their CLP bits are set (i.e., they are sent out-of-rate).

Destination Rule 6: An out-of-rate FRM cell may be turned around either in-rate (with CLP=0) or out-of-rate (with CLP=1).

2.7 Switch Behavior Rules

The switch behavior specifies that the switch must implement some form of congestion control, and rules regarding processing, queuing and generation of RM cells.

Switch Rule 1: This rule specifies that one or more methods of feedback marking methods must be implemented at the switch. The possible methods include:

- **EFCI Marking:** This defines the binary (bit-based) feedback framework, where switches may set the EFCI bit in data cell headers. Note that the VC's EFCI state at the destination is set and reset whenever an incoming data cell has its EFCI set or reset respectively.
- **Relative Rate Marking:** This option allows the switch to set two bits in the RM cell, which have a specific meaning to when they reach the source end systems. When the CI bit is set, it asks the source to decrease, while the NI bit tells the source not to increase beyond its current rate, ACR. Observe that the source rate may

be further reduced using the explicit rate indication field. These bits allow the switches some more flexibility than the EFCI bit marking. Specifically, the switches can avoid the "beat-down" fairness problem seen in EFCI marking scenarios. The problem occurs because connections going through several switches have a higher probability of their EFCI bits being set, than connections going through a smaller number of switches.

- **Explicit Rate Marking:** Allows the switch to specify exactly what rate it wants a source to send at. To ensure coordination among multiple switches in a connection's path, the switch may reduce (but not increase) the ER field in the RM-cells (in the forward and/or backward directions). This thesis deals mainly with explicit rate feedback from switches.
- **VS/VD Control:** In this mode, the switch may segment the ABR control loop by appearing as a "virtual source" to one side of the loop and as a "virtual destination" to the other side.

Switch Rule 2: This rule specifies how a switch may generate an RM cell in case it is heavily congested and doesn't see RM cells from the source. Basically, the rule allows such RM cells to only decrease the source rate, and these RM cells are sent out-of-rate. The rate of these backward RM-cells shall be limited to 10 cells/second, per connection. When a switch generates an RM-cell it shall set either CI=1 or NI=1, shall set BN=1, and shall set the direction to backward.

Switch Rule 3: This rule says that the RM cells may be transmitted out-of-sequence, but the sequence integrity must be maintained. This rule allows the switch the flexibility to put the RM cells on a priority queue for faster feedback to sources when congested. However, by queuing RM cells separately from the data stream, the correlation between the quantities declared RM cells and the actual values in the data stream might be lost.

Switch Rule 4 and 5: Rule 4 specifies alignment with ITU-T's I.371 draft, and ensures the integrity of the MCR field in the RM cell. Rule 5 allows the optional implementation of a use-it-or-lose-it policy at the switch [9].

2.8 Switch Rate Calculation Algorithms Selection Criteria

ATM Forum has specified the way the feedback information is conveyed to the ABR sources. The standard specifies the source and destination behavior and several fields of RM-cells that can be used by the switches to control source rates. The exact algorithm used by the switches to make use of these fields is implementation specific [10].

A number of congestion schemes were presented. To sort out these proposals, the group decided first to agree on a set of selection criteria. Since these criteria are of general interest and apply to non-ATM networks as well, we describe some of them briefly here [12].

2.8.1 Fairness

The most critical component of fair rate allocation is to define a fair rate allocation policy. A number of fairness policies are possible. A commonly used fairness criterion is *max-min faimess*. However, this definition is unambiguous only if no ABR connections receive bandwidth guarantees (MCR = 0). In the case of nonzero MCRs, various fairness criteria exist [4]. In this thesis, the max-min fairness definition is considered.

The max-min criterion attempts to equally allocate the available bandwidth among all connections bottlenecked at the link. This principle is fair since all connections that share a link obtain an equal share of link bandwidth provided they can all use that fair share, and the only factor which prevents a connection from obtaining higher allocation is its bottleneck link. Moreover, this principle is efficient in that it maximizes the throughput for all connections that have the minimum allocations in the network. The fair share can be computed as follows:

$$FairShare = \frac{C_l - \sum Rates\ of\ Connections\ bottlenecked\ elsewhere}{N_l - \sum Connections\ bottlenecked\ elsewhere} \tag{2.3}$$

Where C_l is the capacity (or bandwidth) of link l, and N_l is the number of connections using link l. A simple procedure for finding the max-min fair rate allocations can be formulated iteratively as follows:

1. Find the equal share for the connections on each link.
2. Find the connection(s) with minimum allocated rate.
3. Subtract this rate at the link and eliminate the connections with minimum allocation.
4. Recompute an equal share of each link in the reduced network.

Repeat procedures 2-4 until all the connections are eliminated.

2.8.2 Convergence

At steady state, a fair allocation algorithm should stabilize to an optimal rate vector from any initial network conditions and without causing large oscillations. Large oscillations generally result in poor link utilization, low throughput, and buffer overflow problems. Therefore, the fair rate allocation algorithm should reduce oscillations around the optimal rate.

2.8.3 Responsiveness

A fair rate allocation algorithm should take the minimum round-trip time to get close to the optimal rate. The ability to provide fast access to the available bandwidth and rapid rate reductions under congestion are some of important features for the design and implementation of a fair rate allocation algorithm.

2.8.4 Simulation Configurations

A number of network configurations were also agreed upon to compare various proposals [10]. Most of these were straightforward serial connection of switches. The most popular one is the so-called "Parking Lot" configuration for studying fairness. The configuration and its name is derived from theatre parking lots, which consist of several parking areas connected via a single exit path as shown in Figure 4. At the end of the show, congestion occurs as cars exiting from each parking area try to join the main exit stream.

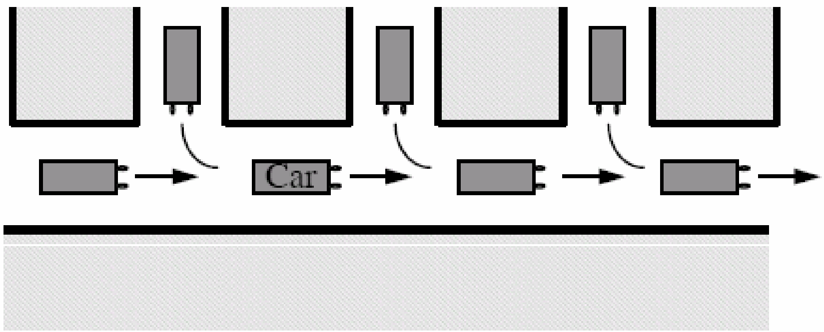

Figure 4 Theatre parking lot

For computer networks, an n-stage parking lot configuration consists of n switches connected in a series. There are n VCs. The first VC starts from the first switch and goes to the end. For the remaining ith VC starts at the i-1th switch. A 3-switch parking lot configuration is shown in Figure 5.

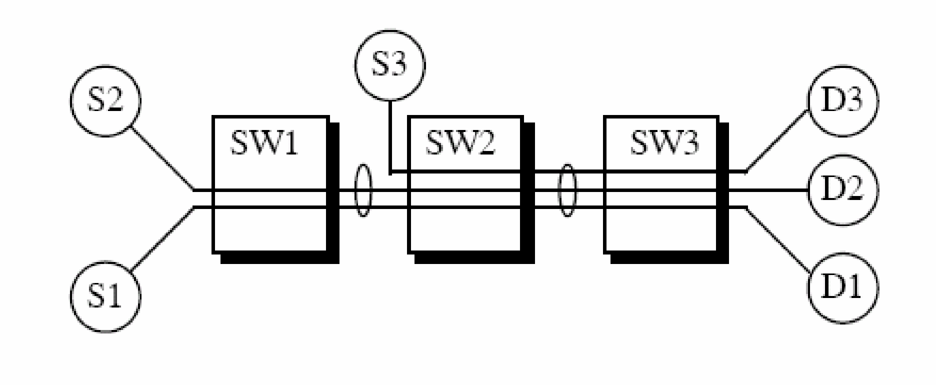

Figure 5 Parking lot configuration

2.8.5 Traffic Patterns

Among the traffic patterns used in various simulations, the following three were most common:

Persistent Sources: These sources, also known as "greedy" or "infinite" sources always have cells to send. Thus, the network is always congested.

Staggered Source: The sources start at different times. This allows us to study the ramp-up (or ramp-down) time of the schemes.

Bursty Sources: These sources oscillate between active and idle state. During active state, they generate a burst of cells.

CHAPTER 3

ERICA ALGORITHM

3.1 Introduction

As we introduced before second-generation switches would perform a fairness algorithm to compute the transmission rate of each connection and use the ER field of the RM cells to convey this rate to the sources. ER switch algorithms have been studied in many contributions [12, 21, 22]. The algorithms attempt to directly compute the fair share. In order to perform this calculation, the switch must maintain a connection-based table to collect and store per-connection information. The algorithms differ in the complexity of implementation in terms of required storage memory and number of floating point divisions required.

Congestion Control with Explicit Rate Indication (CCERI) algorithm was an early proposal to compute explicit rate in a distributed manner, originally formulated in the context of packet switching. The development of this algorithm has had a significant influence on the fair rate allocation algorithms for ABR service. At the time of its development, much of the ABR specification did not exist. Thus, many of the features available now in RM cells were not utilized in its design. In this algorithm, each switch monitors its traffic and calculates its available capacity per flow or per connection. This quantity is called the advertised rate. The switch keeps track of the number of bottlenecked connections at the switch and the last seen ER values. When an RM cell arrives at the switch, if its ER is less than the advertised rate, then the associated connection is assumed to be bottlenecked elsewhere. A bottleneck bit is marked and the current rate of the connection is stored in a connection table. At each iteration, the advertised rate is computed. If at any time a connection previously marked transmits at a rate larger than the advertised rate, the corresponding bottleneck bit is then unmarked, and the advertised rate is recalculated.

The Efficient Rate Allocation Algorithm (ERAA) attempts to solve the computational complexity of the CCERI algorithm. Similar to that algorithm, the switch maintains per-connection information such as bottleneck state and bottleneck bandwidth of each connection. The switch calculates an equal share bandwidth (EQB). When a connection does not use its equal share bandwidth, the connection is considered to be bottlenecked elsewhere. The sum of free bandwidth not used by the bottlenecked connections is referred to as the free bandwidth (FE). A portion of the FB is allocated to the nonbottlenecked connections in addition to their EQB.

Fast Max-Min Rate Allocation Algorithm (FMMRA) is based on measurement of available capacity and exact calculation of fair rates. The goals of this algorithm are to reduce the computational complexity imposed by the CCERI algorithm discussed earlier, and at the same time make the algorithm interoperable. An additional important feature of this algorithm is that it is not sensitive to inaccuracies in CCR values.

ERICA switch algorithm is a well-known algorithm that was extensively studied in the literature and has been used as a point of reference in many other proposals. This algorithm is well understood and found to be efficient. The ERICA algorithm is concerned with the fair and efficient allocation of the available bandwidth to all contending sources. Like any dynamic resource algorithm, it requires monitoring the available capacity and the current demand on the resources. Here, the key "resource" is the available bandwidth at a queuing point (input or output port). In most switches, output buffering is used, which means that most of the queuing happens at the output ports. Thus, ERICA algorithm is applied to each output port (or link)[11].

We present the basic features of the algorithm and explain their operation.

3.2 Basic Algorithm

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

$$z \leftarrow \frac{ABRInputRate}{ABRCapacity} \tag{3.1}$$

where,

$$ABRCapacity \leftarrow T\arg etUtilization(U) * LinkBandwidth \tag{3.2}$$

The input rate is measured over an interval called the switch-averaging interval. The above steps are executed at the end of the switch-averaging interval.

Target utilization (U) is a parameter, which is set to a fraction (close to, but less than 100 %) of the available capacity. Typical values of target utilization are 0.9 and 0.95.

The load factor, z, is an indicator of the congestion level of the link. High overload values are undesirable because they indicate excessive congestion; so are low overload values, which indicate link underutilization. The optimal operating point is at an overload value equal to one. The goal of the switch is to maintain the network at near unit overload.

The fair share of each VC, Fair Share, is also computed as follows:

$$FairShare \leftarrow \frac{ABRCapacity}{NumberofActiveSources} \tag{3.3}$$

The switch allows each source sending at a rate below the Fair Share to rise to Fair Share every time it sends a feedback to the source. If the source does not use all of its Fair Share, then the switch fairly allocates the remaining capacity to the sources, which can use it. For this purpose, the switch calculates the quantity:

$$VCShare \leftarrow \frac{CCR}{z} \tag{3.4}$$

If all VCs changed their rate to their VCShare values then, in the next cycle, the switch would experience unit overload (z equals one). Hence VC Share aims at bringing the system to an efficient operating point, which may not necessarily be fair, and FairShare allocation aims at ensuring fairness, possibly leading to overload (inefficient operation). A combination of these two quantities is used to rapidly reach optimal operation as follows:

$$ERCalculated \leftarrow Max(FairShare, VCShare) \tag{3.5}$$

Sources are allowed to send at a rate of at least FairShare within the first round-trip. This ensures minimum fairness between sources. If the VCShare value is greater than the FairShare value, the source is allowed to send at VCShare, so that the link is not underutilized. This step also allows an unconstrained source to proceed towards its max-min rate. The previous step is one of the key innovations of the ERICA scheme because it improves fairness at every step, even under overload conditions.

The calculated ER value cannot be greater than the ABR Capacity, which has been measured earlier. Hence, we have:

$$ERCalculated \leftarrow Min(ERCalculated, ABRCapacity) \tag{3.6}$$

Since every output port is a queuing point through which a VC passes, every source ought to send at no more than the ER calculated at its bottleneck queuing point. To ensure that the bottleneck ER reaches the source, each switch computes the minimum of the ER it has calculated as above and the ER value in the RM cell. This value is inserted in the ER field of the RM cell.

$$ERCalculated \leftarrow Min(ERinRMcell, ERCalculated) \tag{3.7}$$

3.3 Achieving Max-Min Fairness

Assuming that the measurements do not suffer from high variance, the above algorithm is sufficient to converge to efficient operation in all cases and to the max-min fair allocations in most cases. The convergence from transient conditions to the desired operating point is rapid, often taking less than a round trip time.

However, some cases are discovered in which the basic algorithm does not converge to max-min fair allocations. This happens if all of the following three conditions are met:

- The load factor *z* becomes one.
- There are some sources, which are bottlenecked elsewhere upstream.
- CCR for all remaining sources is greater than the *FairShare*.

If this happens, then the system remains in its current state. This final state may or may not be fair in the max-min sense.

To achieve max-min fairness, the basic ERICA algorithm is extended by remembering the highest allocation made during one averaging interval and ensuring that all eligible sources can also get this high allocation. To do this, a variable is added; MaxAllocPrevious, which stores the maximum allocation given in the previous interval, and another variable MaxAllocCurrent, which accumulates the maximum allocation given during the current switch-averaging interval. Basically, for $z > 1 + \delta$, where δ is a small fraction, we use the basic ERICA algorithm and allocate the source Max (FairShare, VCShare). But, for z <=

$1+\delta$, we attempt to make all the rate allocations equal. We calculate the ER as Max (FairShare, VCShare, MaxAllocPrevious).

3.4 ABR Operation with VBR and CBR in the Background

The discussion so far assumed that the entire link was being shared by ABR sources. Normally, ATM links will be used by constant bit rate (CBR) and variable bit rate (VBR) traffic along with ABR traffic. In fact, CBR and VBR have a higher priority. Only the capacity left unused by VBR and CBR is given out to ABR sources. For such links, we need to measure the CBR and VBR usage along with the input rate. The ABR capacity is then calculated as follows:

$$ABRCapacity \leftarrow T\arg etUtilization * LinkBandwidth - VBRUsage - CBRUsage \quad (3.8)$$

The rest of ERICA algorithm remains unchanged. Notice that the target utilization is applied to the entire link bandwidth and not the left over capacity. That is,

$$ABRCapacity \neq T\arg etUtilization * \{LinkBandwidth - VBRUsage - CBRUsage\} \quad (3.9)$$

3.5 Flow Chart of the Algorithm

A complete flow chart of the algorithm is presented as follows. The flow chart shows steps to be taken on three possible events: at the end of an averaging interval, on receiving a cell (data or RM), and on receiving a backward RM cell.

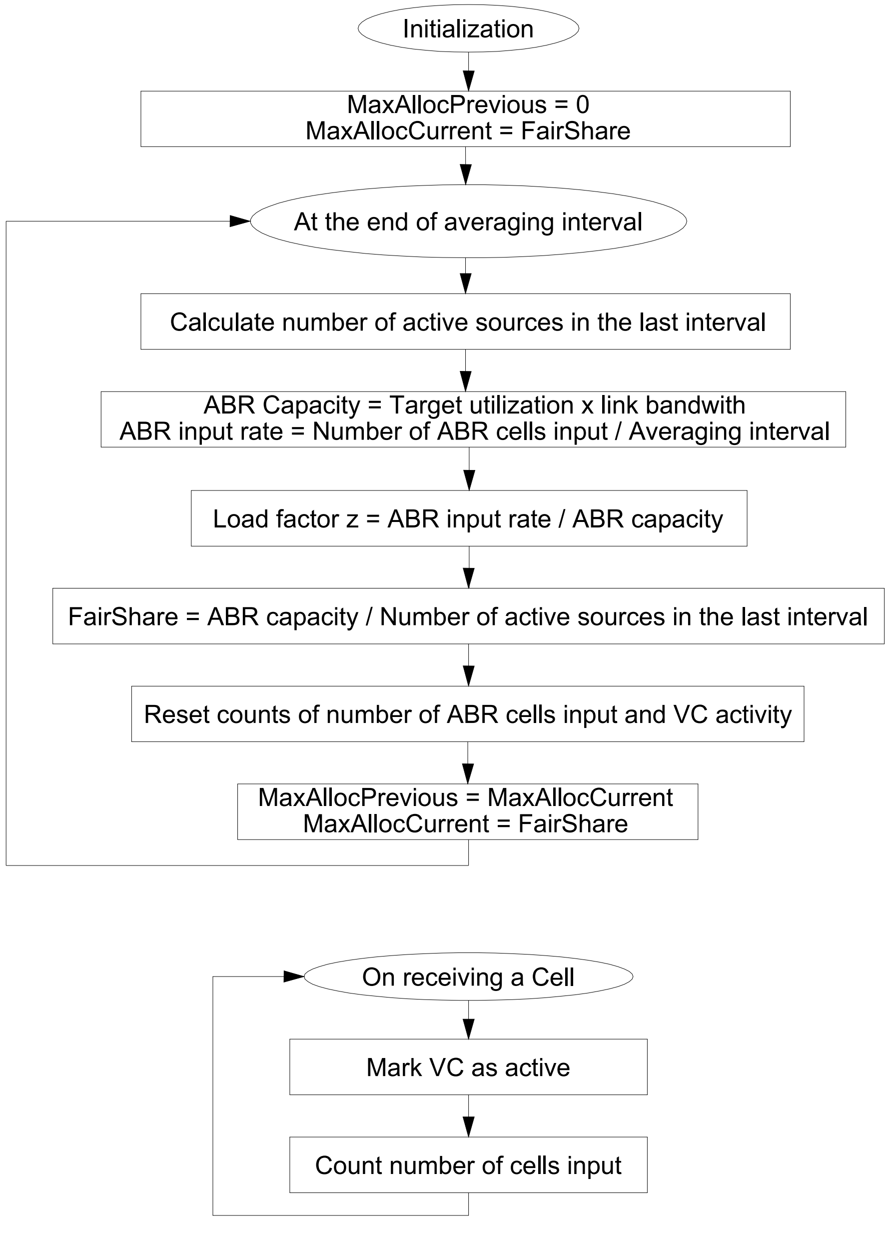

Figure 6 Flow chart of at the end of averaging interval and on receiving a cell

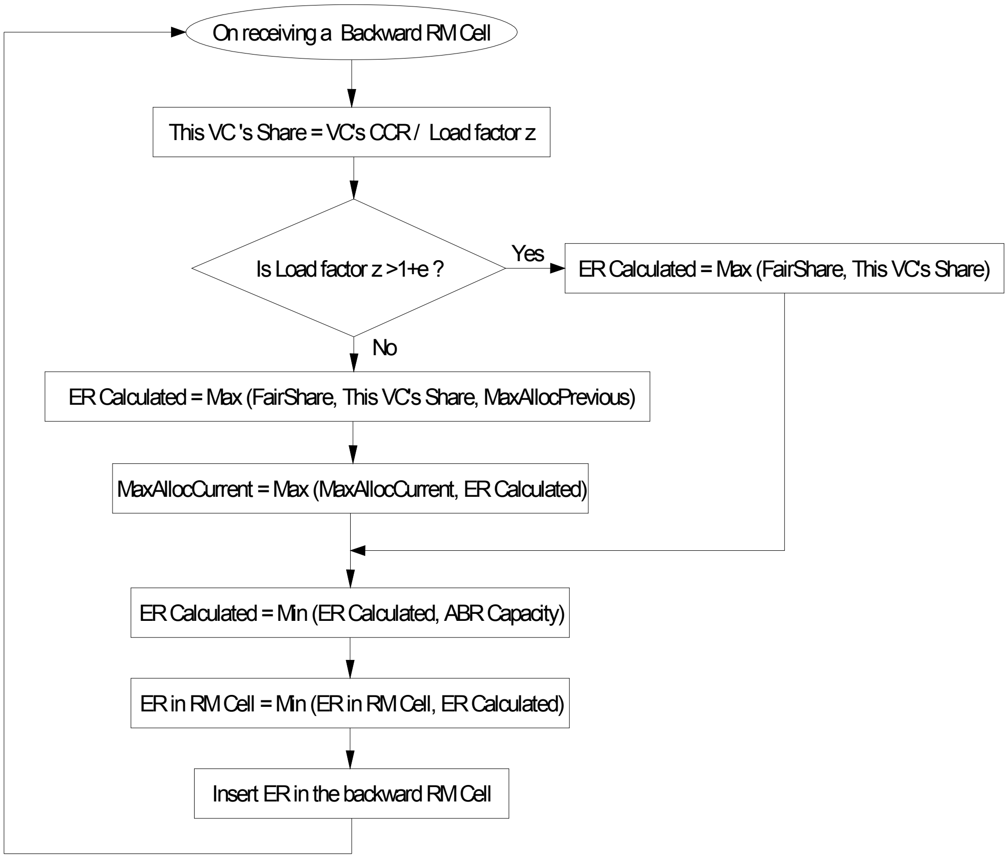

Figure 7 Flow chart of on receiving backward RM cell

CHAPTER 4

FORMULATION OF CONGESTION CONTROL AS A CONTROL PROBLEM

4.1 Introduction

Early implementation of congestion control was based on binary feedback information where switches only mark a single congestion bit in the data cells in the event of congestion. The inherent shortcomings of the binary approach have recently led to the consideration and implementation of sophisticated fair rate allocation algorithms that compute fair rates for each connection and convey this information to the sources [12].

In fair rate allocation algorithms the node measures the average rate available to ABR sources at periodic intervals and simply divides a fraction of this capacity equally among the various users. This is the basic approach used in [4] and has been examined in the previous chapter.

In this chapter, to compute data rates for ABR sources we use both available bandwidth and queue length and introduce a control-based mathematical model that helps us to implement an explicit-rate congestion control algorithm.

4.2 Problem Formulation

The model adopted here is a discrete-time model, where a time unit corresponds to the interval over which the rate available to ABR sources and queue length values are determined.

Consider a graph of communication links shared by a number of connections (source-destination pairs)[19].

$L = \{1, 2, ..., l\}$: Set of links in the network

$S = \{1, 2, ..., s\}$: Set of connections using these links

r_s: Flow rate of each connection

x_s: Flow rate of each connection, which does not include guaranteed minimum rate

$L_s \subset L$: Set of links used by connection s

$S_l \subset S$: Set of connections using link l

C_l: Capacity of link l

MR_s: Guaranteed minimum rate of each connection which could be zero

F_l: Total flow on link l

According to these notations flow rate of each ABR connection should be equal to $r_s := x_s + MR_s$. Then the aggregate flow on link l can be written as

$$F_l = \sum_{s \in S_l} r_s = \sum_{s \in S_l} x_s + G_l \qquad (4.1)$$

where $G_l = \sum_{s \in S_l} MR_s$.

We have the following constraints on the vector $x := \{x_s \mid s \in S\}$ of flow rates:

$$\begin{aligned} & x_s \geq 0, \forall s \in S \\ & F_l \leq C_l, \forall l \in L \end{aligned} \tag{4.2}$$

Note that for a given vector $C := \{C_l \mid l \in L\}$ of capacities, no vector x of rates might exist if $G_l > C_l$ for any l. Therefore, we further assume that the vector C of capacities satisfies

$$C_l \geq G_l, \forall l \in L \tag{4.3}$$

A pair of vectors (x, C) satisfying these constraints is said to be *feasible.*

One of the goals of congestion control is to treat all connections *fairly.* The most widely accepted notion of fairness is *max-min fairness.* A vector of rates x is said to be max-min fair if it is feasible and for each $s \in S, x_s$ cannot be increased while maintaining feasibility without decreasing $x_s^{'}$ for some connection $s^{'}$ for which $x_s^{'} \leq x_s$.

Given a feasible rate vector x, we say that a link is a bottleneck link with respect to x for a connection s crossing l if $C_l = F_l$ and $x_s^{'} \leq x_s$ for all connections $s^{'}$ crossing link l. It can be shown that a feasible rate vector x is max-min fair if and only if each connection has a bottleneck link with respect to x. Hence, in a max-min fair allocation each connection necessarily has a bottleneck link. Now, consider a bottleneck link l in the network. By the definition of a bottleneck link we must have $C_l = F_l$. The set S_l of connections crossing l can be partitioned into two disjoint subsets, B_l and B_l^c, where ^{c} is the complement operation. Here B_l denotes the set of connections that have l as a bottleneck link. By definition, for any two connections $p, q \in B_l$, we have $x_p \geq x_q$ and $x_q \geq x_p$ for all $s \in S$. In particular, $x_p \geq x_q$ and $x_q \geq x_p$, which implies that $x_q = x_p$. In other words, to have a max-min fair allocation, all connections bottlenecked on a particular link must have the same flow rate. This observation yields itself to an alternative definition of max-min fairness provided that the number of bottlenecked connections on a given link is known. Under this assumption, the max-min fair share of a connection bottlenecked on link l should be equal to

$$r_s = MR_s + \frac{C_l - (u_l + G_l)}{M_{l,l(\infty)}}, \forall s \in B_l \tag{4.4}$$

where u_l is the sum of the rates of connections that are not bottlenecked on link l, and $M_{l,l}(\infty)$ is the number of connections bottlenecked on link l.

High utilization of the resources and a low loss rate are also the objectives of congestion control. These objectives can be achieved by regulating the queue length at switches around a target level. Tracking such a nominal queue length is desirable in order to avoid losses due to overflow and waste of the link capacity due to underflow. Let us give some other formulations related to the queue length.

q_l : Size of the queue at the switch output port-controlling link l.

Q_l : Size of the link buffer

The value of the q_l can be maximum buffer capacity. So it should satisfy,

$$0 \leq q_l \leq Q_l \tag{4.5}$$

Now, given this network setting, starting with arbitrary initial vectors $r = \{r_s \mid s \in S\}$ of rates, and $q = \{q_l \mid l \in L\}$ of queue sizes, we want the pair (r, q) to converge to (r^*, q^*), where r^* is a vector of max-min fair rates, and q^* is the vector of desired queue lengths. In the next sub-section we describe an explicit-rate congestion control algorithm, where connection s updates its own rate r_s based on a limited amount of information from the network and is also easy-to-implement.

4.3 The Algorithm

Here, using some simplifying assumptions an easy-to-implement globally convergent explicit-rate congestion control algorithm is presented.

$r(n)$: Total number of packets that arrive at the link in the interval $[n, n+1)$

$a(n)$: Total number of packets that depart from the link buffer in the interval $[n, n+1)$

$q(n)$: Queue length at time n

$\bar{Q}$: Size of the link buffer

Queue length $q(n)$ evolves according to

$$q(n+1) = \min\{\bar{Q}, \max\{0, q(n) + r(n) - a(n)\}\} \tag{4.6}$$

Here min function is used to ensure that queue length value is always equal or smaller than queue buffer size. Max function puts a lower bound to queue length, which should not be negative.

N: Total number of connections crossing l, which should $N \geq 1$

$r_m(n)$: Number of packets that arrive from source m during time slot $[n, n+1)$

We can write $r(n)$ as:

$$r(n) = \sum_{m=1}^{N} r_m(n) \tag{4.7}$$

$r(n)$ has two components:

The number of packets that arrive from uncontrolled sources, i.e., those sources, which are bottlenecked elsewhere in the network or are limited by their peak cell rate constraints. The ER controller at this switch has no control over these sources. This component is denoted by $u(n)$.

The number of packets that arrive from controlled sources, which are bottlenecked at this switch.

M : Total number of controlled sources, which should satisfies $M \geq 1$

$u(n)$: Number of packets that arrive from uncontrolled sources in the interval $[n, n+1)$

r_m^c : Number of packets that arrive from controlled source m in the interval $[n, n+1)$

$$r(n) = \sum_{m=1}^{M} r_m^c(n) + u(n) \tag{4.8}$$

Number of controlled sources, M may not be known to the switch.

It takes time from the moment the ER decision is made by the switch until an action is taken by a source, until subsequently that action affects the state of the switch that initiated this action. Thus, $r_m(n)$ is actually an outcome of an action taken d_m time units earlier, where d_m represents the action delay for source m and is taken to be independent of time n. Thus, we have

$$r_m^c(n) = MR_m + ER(n - d_m) \tag{4.9}$$

where $ER(n)$ denotes the action of the switch (ER controller) at time n. Without any loss of generality, we assume that the d_m 's are ordered such that

$$0 \le d_1 \le \ldots \le d_M \le \bar{d}$$

where $\bar{d}$ corresponds to the maximum network delay.

The action delay, d_m, for source m, consists of several components, such as the round-trip propagation delay, the queuing and processing delays, etc. Since queuing and processing delays are subject to variation, it is impossible for a link level controller to predict the exact value of d_m. Further, any calculation of the fair share at the switch requires the knowledge of the number of controlled connections, $M\ (= N - M^u)$, which is not known to the switch either. Motivated by these observations, we want to develop a robust explicit rate congestion control algorithm, which does not require M.

Note that queue length at time n can be written in the following form:

$$q(n+1) = \min\{\bar{Q}, \max\{0, q(n) + \overline{MR} + \sum_{k=0}^{\bar{d}} m_k ER(n-k) + u(n) - a(n)\}\} \tag{4.10}$$

$\overline{MR}$: Sum of MR 's of active connections

m_k : Number of controlled connections having k units of action delay.

Even though the available bandwidth and aggregate rate of uncontrolled sources on the link may vary, to focus on the derivation of the algorithm, we assume that these quantities do not change with time.

In order to simplify the final design, we also want to ease the effect of multiple time delays on the system dynamics. One way for the switch to do this is to wait long enough after issuing the explicit rate, so that all connections adjust their rates to issued rate. As the switch has an estimate of the round-trip delay of each connection on its link, putting an upper bound, $\overline{d}$, on the maximum round-trip delay is feasible. Thus, if we update $ER(n)$ every $(\overline{d}+1)$ time units, all controlled sources will have enough time to modify their transmission rates according to the explicit rate fed back at the previous update interval.

For ease of notation, let us introduce the following subsequences:

$$q_s(n) = q(n(\overline{d}+1)),\ ER_s(n) = ER(n(\overline{d}+1)), r_s(n) = r(n(\overline{d}+1))$$

Note that $ER(n)$ is kept at the same value for an interval of length $(\overline{d}+1)$. That is,

$$ER(n(\overline{d}+1)) = ER(n(\overline{d}+1)+1) = \ldots = ER_l((n+1)(\overline{d}+1)-1)$$

To achieve the dual goal of max-min fairness and queue length stability, $ER_s(n)$ will be updated according to

$$ER_s(n) = \min\{a, \max\{0, ER_s(n-1) - \alpha(r_s(n)-a) - \beta(q_s(n)-Q)\}\} \qquad (4.11)$$

$$ER_s(0) = a$$

where Q is the target queue length, a is the available bandwidth and α and β are parameters to be selected to meet various design criteria. Here the max function is introduced to ensure that the switch asks the source to transmit at a positive rate in excess of the minimum rate and the min function puts an upper bound on the maximum allowed rate for the sources. In (4.11), the term $-\beta(q_s(n)-Q)$ is introduced to drive the queue length to the desired set point by providing negative feedback in the closed-loop system dynamics. Note that, this algorithm does not require any per-flow information, as it only uses the aggregate flow $r_s(n)$ on link l.

If $q_s(n)$ converges to Q, $ER_s(n)$ converges to

$$ER_s(\infty) = \frac{a-(u+\overline{MR})}{M}$$

since $r_s(n) = u + \overline{MR} + MER_s(n-1)$. If we can show that $q_s(n) \to Q$, then by (4.9), the rate of controlled source m goes to

$$r_m^c(\infty) = MR_m + ER_s(\infty) = MR_m + \frac{a-(u+\overline{MR})}{M} \tag{4.12}$$

If the controller gains (α, β) are picked properly, and if the size of the buffer, $\bar{Q}$, is large enough this algorithm is indeed globally asymptotically stable.

4.4 Analysis of the Algorithm

4.4.1 State Space Formulation

We introduce shifted variables as:

$$\begin{aligned} x(n) &= q(n) - Q \\ x_s(n) &= q_s(n) - Q \end{aligned}$$

First we write the queue length according to (4.10), and then rewrite this equation in term of newly introduced variables,

$$q(n+1) = \min\{\bar{Q}, \max\{0, q(n) + \overline{MR} + \sum_{k=0}^{\bar{d}} m_k ER(n-k) + u(n) - a(n)\}\}$$

$$x(n+1) = q(n+1) - Q$$

$$x(n+1) = \min\{\bar{Q} - Q, \max\{-Q, x(n) + \overline{MR} + \sum_{k=0}^{\bar{d}} m_k ER(n-k) + u - a\}\} \qquad (4.13)$$

$$ER_s(n) = \min\{a, \max\{0, ER_s(n-1) - \alpha(r_s(n) - a) - \beta(q_s(n) - Q)\}\}$$

Since $r_s(n) = u + \overline{MR} + MER_s(n-1)$ and $x_s(n) = q_s(n) - Q$, $ER_s(n)$ can be written as:

$$ER_s(n) = \min\{a, \max\{0, ER_s(n-1) - \alpha(MER_s(n-1) + \overline{MR} + u - a) - \beta x_s(n)\}\} \quad (4.14)$$

Let us consider the evaluation of $x(n)$ for $(\bar{d}+1)$ time units starting from $n(\bar{d}+1), n \geq 0$. We have

$$
\begin{aligned}
x(n(\bar{d}+1)+1) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)) + \overline{MR} - a + u \\
&+ \sum_{k=0}^{\bar{d}} m_k ER(n(\bar{d}+1)-k)\}\} \\
x(n(\bar{d}+1)+1) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)) + \overline{MR} - a + u \\
&+ m_0 ER(n(\bar{d}+1)) + m_1 ER(n(\bar{d}+1)-1) + \sum_{k=2}^{\bar{d}} m_k ER(n(\bar{d}+1)-k)\}\} \\
x(n(\bar{d}+1)+1) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)) + \overline{MR} - a + u \\
&+ m_0 ER_s(n) + (M-m_0) ER_s(n-1)\}\}
\end{aligned}
$$

$$
\begin{aligned}
x(n(\bar{d}+1)+2) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)+1) + \overline{MR} - a + u \\
&+ \sum_{k=0}^{\bar{d}} m_k ER(n(\bar{d}+1)+1-k)\}\} \\
x(n(\bar{d}+1)+2) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)+1) + \overline{MR} - a + u \\
&+ m_0 ER(n(\bar{d}+1)+1) + m_1 ER(n(\bar{d}+1)) + \sum_{k=2}^{\bar{d}} m_k ER(n(\bar{d}+1)+1-k)\}\} \\
x(n(\bar{d}+1)+1) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)) + \overline{MR} - a + u \\
&+ (m_0+m_1) ER_s(n) + (M-m_0-m_1) ER_s(n-1)\}\}
\end{aligned}
$$

$$
\begin{aligned}
x((n+1)(\bar{d}+1)) = \min\{&\bar{Q}-Q, \max\{-Q, x((n+1)(\bar{d}+1)-1) + \overline{MR} - a + u \\
&+ \sum_{k=0}^{\bar{d}} m_k ER((n+1)(\bar{d}+1)-1-k)\}\} \qquad (4.15) \\
x((n+1)(\bar{d}+1)) = \min\{&\bar{Q}-Q, \max\{-Q, x((n+1)(\bar{d}+1)-1) + \overline{MR} - a + u \\
&+ m_0 ER((n+1)(\bar{d}+1)-1) + m_1 ER((n+1)(\bar{d}+1)-2) + \sum_{k=2}^{\bar{d}} m_k ER((n+1)(\bar{d}+1)-1-k)\}\} \\
x((n+1)(\bar{d}+1)) = \min\{&\bar{Q}-Q, \max\{-Q, x((n+1)(\bar{d}+1)-1) + \overline{MR} - a + u \\
&+ MER_s(n)\}\}
\end{aligned}
$$

where $0 \le m_k \le M$, $k = 1, \ldots, \bar{d}$. Furthermore $\sum_{k=0}^{\bar{d}} m_k = M$.

Now we know that,

$$
\begin{aligned}
x(n(\bar{d}+1)+1) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)) + \overline{MR} - a + u \\
&+ m_0 ER_s(n) + (M-m_0) ER_s(n-1)\}\}
\end{aligned}
$$

$$
\begin{aligned}
x(n(\bar{d}+1)+2) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)+1) + \overline{MR} - a + u \\
&+ (m_0+m_1) ER_s(n) + (M-m_0-m_1) ER_s(n-1)\}\}
\end{aligned}
$$

We rewrite $x(n(\bar{d}+1)+2)$ in terms of $x(n(\bar{d}+1)+1)$ as:

$$
\begin{aligned}
x(n(\bar{d}+1)+2) = \min\{&\bar{Q}-Q, \max\{-Q, x(n(\bar{d}+1)) + \overline{MR} - a + u \\
&+ m_0 ER_s(n) + (M-m_0) ER_s(n-1) + \overline{MR} - a + u \\
&+ (m_0+m_1) ER_s(n) + (M-m_0-m_1) ER_s(n-1)\}\}
\end{aligned}
$$

$$
\begin{aligned}
= \min\{&\bar{Q}-Q, \max\{-Q, x_s(n) + \gamma_0 ER_s(n) + \gamma_1 ER_s(n-1) \\
&+ 2u - 2a + 2\overline{MR}\}\}
\end{aligned}
$$

Note that for this substitution

$$
\begin{aligned}
&\gamma_0 = m_0 + m_0 + m_1 \\
&\gamma_1 = (M-m_0) + (M-m_0-m_1) \\
&\gamma_0 + \gamma_1 = 2M
\end{aligned}
$$

If we recursively substitute equations (4.15) into one another starting from the top one, we can express $x_s(n+1)$ in terms of $x_s(n)$, $ER_s(n)$ and $ER_s(n-1)$.

$$
x_s(n+1) = \min\{\bar{Q}-Q, \max\{-Q, x_s(n) + \gamma_0 ER_s(n) + \gamma_1 ER_s(n-1) + \frac{\gamma_0+\gamma_1}{M}(u + \overline{MR} - a)\}\} \tag{4.16}
$$

where

$$
1 \le \gamma_0 \le \bar{\gamma}_0 := \sum_{k=0}^{\bar{d}} (\bar{d}+1-k) m_k \tag{4.17}
$$

$$
1 \le \gamma_1 \le \bar{\gamma}_1 := \sum_{k=0}^{\bar{d}} k m_k \tag{4.18}
$$

where, in general $\gamma_0 + \gamma_1 = kM$, for some $1 \le k \le (\bar{d}+1)$. Note that $\bar{\gamma}_0$ and $\bar{\gamma}_1$ satisfy $\bar{\gamma}_0 + \bar{\gamma}_1 = (\bar{d}+1)M$.

Let $e := u + \overline{MR}$. The original system (4.13)-(4.14) can be written in terms of $x_s(n)$ as follows:

$$x_s(n+1) = \min\{\bar{Q}-Q, \max\{-Q, x_s(n) + \gamma_0 ER_s(n) + \gamma_1 ER_s(n-1) + \frac{\gamma_0+\gamma_1}{M}(e-a)\}\} \quad (4.19)$$

$$ER_s(n) = \min\{a, \max\{0, (1-\alpha M)ER_s(n-1) - \alpha(e-a) - \beta x_s(n)\}\} \qquad (4.20)$$

Claim:

The saturation nonlinearities imposed on $ER_s(n)$ are redundant provided that α and β are picked properly. In other words if $0 \le ER_s(n) \le a$, which is our case, then $0 \le ER_s(n) \le a$ for all $n \ge 1$, which in turn implies that saturation nonlinearities in (4.20) never become active. Let $y(n)$ denote the linear version of the right-hand side of (4.20), i.e.

$$y(n) := (1-\alpha M)y(n-1) - \alpha(e-a) - \beta x_s(n) \qquad (4.21)$$

and assume that

$$0 \le \alpha \le \frac{1}{M} \qquad (4.22)$$

and

$$0 \le \beta \le \min\{\frac{\alpha(a-e)}{\bar{Q}-Q}, \frac{\alpha[(M-1)a+e]}{Q}\} \qquad (4.23)$$

Proof by induction is applied

Claim is true for $n=0$, i.e., $0 \le y(0) \le a$.

We suppose claim is true for $k_0 \ge 0$, i.e., $0 \le y(k_0) \le a$

We will show that whether the claim is also true for $k_0 + 1 \ge 0$, i.e., $0 \le y(k_0+1) \le a$ is also true or not.

$$y(k_0+1) := (1-\alpha M)y(k_0) - \alpha(e-a) - \beta x_s(k_0+1)$$

Since $x_s(n) \le \bar{Q} - Q \quad \Rightarrow \quad -x_s(n) \ge -(\bar{Q} - Q)$

We can write $y(k_0+1)$ as:

$$y(k_0+1) \ge (1-\alpha M)y(k_0) - \alpha(e-a) - \beta(\bar{Q} - Q)$$

Since $y(k_0) \ge 0$

$$y(k_0+1) \ge -\alpha(e-a) - \beta(\bar{Q} - Q)$$

Since,

$$\begin{aligned}
&0 \le \beta \le \min\{\frac{\alpha(a-e)}{\bar{Q}-Q}, \frac{\alpha[(M-1)a+e]}{Q}\} \\
&\Leftrightarrow 0 \le \beta \le \frac{\alpha(a-e)}{\bar{Q}-Q} \\
&\Leftrightarrow (\bar{Q}-Q)\beta \le \alpha(a-e) \\
&\Leftrightarrow 0 \le \alpha(a-e) - \beta(\bar{Q}-Q)
\end{aligned}$$

We write $y(k_0+1)$ as:

$$y(k_0+1) \ge 0$$

This is the proof of $y(k_0+1) \ge 0$. For the upper bound of $y(k_0+1)$, consider

$$y(k_0+1) := (1-\alpha M)y(k_0) - \alpha(e-a) - \beta x_s(k_0+1)$$

Since $x_s(n) \ge -Q \quad \Leftrightarrow \quad -x_s(n) \le Q \quad \Leftrightarrow \quad -\beta x_s(n) \le \beta Q$

We can write $y(k_0+1)$ as:

$$y(k_0+1) \le (1-\alpha M)y(k_0) - \alpha(e-a) + \beta Q$$

Since

$$\beta \le \frac{\alpha[(M-1)a+e]}{Q} \quad and \quad y(k_0) \le a$$

$y(k_0+1)$ can be written as

$$y(k_0+1) \le (1-\alpha M)a - \alpha(e-a) + \frac{\alpha[(M-1)a+e]}{Q}Q$$

$$y(k_0+1) \le a$$

Combining these two inequalities we obtain

$$0 \le y(k_0+1) \le a$$

Since k_0 is arbitrary, we have $0 \le y(n) \le a$ for all $n \ge 1$, provided that $0 \le y(0) \le a$ and the pair (α, β) is picked accordance with (4.22)-(4.23). Consequently, we can remove the min and max functions from (4.20), and substitute for $ER_s(n)$ in (4.19). This results in

$$\begin{aligned} x_s(n+1) = \min\{\bar{Q} - Q, \max\{-Q, (1-\beta\gamma_0)x_s(n) + \\ [\gamma_0(1-\alpha M)+\gamma_1]ER_s(n-1) - \gamma_0\alpha(e-a) + \frac{\gamma_0+\gamma_1}{M}(e-a)\}\} \end{aligned} \tag{4.24}$$

Now, we are in position to write the system dynamics in the state-space form. Introducing the state variables

$$x_0(n) := x_s(n)$$

$$x_1(n) := ER_s(n-1) - \frac{a-e}{M}$$

We want to find the value of $[\gamma_0(1-\alpha M)+\gamma_1]ER_s(n-1)$ in terms of $x_1(n)$.

$$[\gamma_0(1-\alpha M)+\gamma_1]x_1(n)=[\gamma_0(1-\alpha M)+\gamma_1]ER_s(n-1)-\frac{a-e}{M}[\gamma_0(1-\alpha M)+\gamma_1]$$

$$[\gamma_0(1-\alpha M)+\gamma_1]ER_s(n-1)=[\gamma_0(1-\alpha M)+\gamma_1]x_1(n)+\frac{a-e}{M}[\gamma_0(1-\alpha M)+\gamma_1]$$

If we substitute $[\gamma_0(1-\alpha M)+\gamma_1]ER_s(n-1)$ in (4.24),

$$x_s(n+1)=\min\{\bar{Q}-Q,\max\{-Q,(1-\beta\gamma_0)x_s(n)+[\gamma_0(1-\alpha M)+\gamma_1]x_1(n)$$
$$+\frac{a-e}{M}[\gamma_0(1-\alpha M)+\gamma_1]-\gamma_0\alpha(e-a)+\frac{\gamma_0+\gamma_1}{M}(e-a)\}\}$$

We obtain

$$x_0(n+1)=x_s(n+1)=\min\{\bar{Q}-Q,\max\{-Q,(1-\beta\gamma_0)x_0(n)+[\gamma_0(1-\alpha M)+\gamma_1]x_1(n)\}\} \quad (4.25)$$

Let us write $x_1(n+1)$ in terms of $x_1(n)$ and $x_0(n)$.

$$x_1(n):=ER_s(n-1)-\frac{a-e}{M}$$
$$x_1(n+1):=ER_s(n)-\frac{a-e}{M}$$

From (4.20) we know that

$$ER_s(n)=(1-\alpha M)ER_s(n-1)-\alpha(e-a)-\beta x_s(n)$$

We can write $x_1(n+1)$ as:

$$x_1(n+1):=(1-\alpha M)ER_s(n-1)-\alpha(e-a)-\beta x_s(n)-\frac{a-e}{M}$$

Since

$$(1-\alpha M)x_1(n):=(1-\alpha M)ER_s(n-1)-\frac{(1-\alpha M)(a-e)}{M}$$

$x_1(n+1)$ is equal to

$$x_1(n+1) := (1-\alpha M)x_1(n) + \frac{(1-\alpha M)(a-e)}{M} - \alpha(e-a) - \beta x_0(n) - \frac{a-e}{M}$$
$$x_1(n+1) := (1-\alpha M)x_1(n) - \beta x_0(n)$$

Since

$$0 \le ER_s(n) \le a \quad \Rightarrow \quad -\frac{a-e}{M} \le x_1(n) \le a - \frac{a-e}{M}$$

We obtain then,

$$x_1(n+1) = \min\{a - \frac{a-e}{M}, \max\{-\frac{a-e}{M}, -\beta x_0(n) + (1-\alpha M)x_1(n)\}\} \qquad (4.26)$$

Hence

$$\overline{x}(n+1) = \begin{pmatrix} x_0(n+1) \\ x_1(n+1) \end{pmatrix} = sat(A\overline{x}(n)) \qquad (4.27)$$

for $\overline{x} = [x_0 x_1]^T \in R^2$, and

$$A = \begin{pmatrix} 1-\beta\gamma_0 & \gamma_0(1-\alpha M) + \gamma_1 \\ -\beta & 1-\alpha M \end{pmatrix} \qquad (4.28)$$

Let $\overline{y} = [y_0\, y_1]^T$. We define the saturation function as $sat(\overline{y}) := [sat_0(y_0)\ sat_1(y_1)] : R^2 \to R^2$, where

$$sat_0(y_0) := \begin{cases} \bar{Q} - Q, & y_0 > \bar{Q} - Q \\ y_{0,} & -Q \leq y_0 \leq \bar{Q} - Q \\ -Q, & y_0 < -Q \end{cases}$$

$$sat_1(y_1) := \begin{cases} a - \frac{a-e}{M}, & y_1 > a - \frac{a-e}{M} \\ y_{1,} & -\frac{a-e}{M} \leq y_1 \leq a - \frac{a-e}{M} \\ -\frac{a-e}{M}, & y_1 < -\frac{a-e}{M} \end{cases}$$

4.4.2 A Robust Global Stability Result

Here, we will show that the class of nonlinear systems (4.25)-(4.26) with an arbitrary pair of $(\gamma_0 \gamma_1)$ satisfying (4.17)-(4.18), is globally asymptotically stable if the controller gains (α, β) are picked appropriately. This result will enable us to conclude that our ER congestion control algorithm achieves max-min fairness along with queue length stability under minimum rate and finite buffer length constraints. Note that the algorithm does not require explicit knowledge of round-trip delays, and it adapts to the variations in the number of controlled connections.

Lemma 1**:** The system (4.27) has unique equilibrium point at the origin.
Lemma 2**:** The linear system corresponding to (4.27)

$$x^l(n+1) = Ax^l(n) \tag{4.29}$$

is globally asymptotically stable, if α and β satisfy

$$0 < \alpha < \frac{1}{M} \tag{4.30}$$

$$0 < \beta < \min\{\frac{1}{\gamma_0}, \frac{\alpha M}{\gamma_1}\} \tag{4.31}$$

Lemma 3: The nonlinear system corresponding to (4.27) is globally asymptotically stable, if $A=[a_{ij}]$ is stable, and the following conditions are satisfied:

$$\begin{aligned} &\left|a_{11}-a_{22}\right| \le 2\lambda+1-\det(A) \\ &Q_{\max} \quad > p_{\max}A_{\min} \\ &Q_{\max}Q_{\min} \quad > p_{\max}A_{\min}A_{\max} \end{aligned}$$

where

$$\begin{aligned} p_{\max} &= \frac{1-\beta\gamma_0}{1-\alpha M}+\frac{(\alpha M-\beta\gamma_1)^2}{2(1-\alpha M)^2}\Bigg(1+\frac{1}{\sqrt{\dfrac{(\alpha M-\beta\gamma_0)^2}{(\alpha M-\beta\gamma_1)^2+4(1-\beta\gamma_0)(1-\alpha M)}}}\Bigg) \\ Q_{\max} &= \max\{\bar{Q}-Q,Q\} \\ Q_{\min} &= \min\{\bar{Q}-Q,Q\} \\ A_{\max} &= \max\{\frac{a-(a-e)}{M},\frac{(a-e)}{M}\} \\ A_{\min} &= \min\{\frac{a-(a-e)}{M},\frac{(a-e)}{M}\} \end{aligned}$$

Corollary:

If $(\alpha,\beta,Q,\bar{Q})$ are selected as

$$0<\alpha<\frac{2}{3M} \tag{4.32}$$

$$Q_{\max} > p_{\max}A_{\min} \tag{4.33}$$

$$Q_{\min}Q_{\max} > p_{\max}A_{\min}A_{\max} \tag{4.34}$$

$$0<\beta<\alpha\min\{\frac{1}{\bar{d}+1},\frac{\alpha(a-e)}{\bar{Q}-Q},\frac{\alpha[(M-1)a+e]}{Q}\} \tag{4.35}$$

then ER control algorithm asymptotically achieves

$$\lim_{n\to\infty}\{q_n(n), ER(n)\} = \left\{Q, \frac{a-(u+\overline{MCR})}{M}\right\}$$

resulting in a max-min fair bandwidth allocation along with a stable queue length.

4.5 Selected Parameters in the Algorithm

In our simulations, control scheme's interval is defined as every 0.001 seconds or 50 cells, whichever comes first. 1 km implies $5\mu\sec$ propagation delay. So our round trip delay is at most $60\mu\sec$ according to the chosen configurations. To receive 50 cells will take more than one round trip delay. This means switches are going to wait long enough after issuing the explicit rate, so that all connections adjust their rates to issued rate.

As we said before if the controller gains (α, β) are picked properly, and if the size of the buffer, $\bar{Q}$, is large enough this algorithm is indeed globally asymptotically stable. The controller gains are picked as (α, β)=(0.1, 0.01) in accordance with the theory developed. In our simulations we use at most five connections sharing a link. We have chosen α such that it satisfies (4.32). In accordance with the equations β and Q values have been chosen as 0.01 and 300 cells respectively.

CHAPTER 5

SIMULATION STUDIES

In this section we evaluate explicit-rate congestion control algorithm implemented with the help of control-based mathematical model and ABR congestion control scheme “ERICA” with respect to some properties, such as efficiency, fairness and responsiveness. The simulation experiments reported here were conducted using an “event-scheduling” simulation approach. This approach is based on the concept that actions may be taken only when one of the events takes place.

In the program the user can create different network topologies to test the congestion control schemes. First you set the parameters of network components and make the connections of these components. While the simulation is running, various instantaneous performance values are written in the text files. Data logging frequency is determined by the user. We are measuring ACR and queue length values of the selected sources and switches. Also the utilization of the bottleneck links is measured.

The network components sending messages to one another. The components include switches, traffic generators and destinations. Traffic generators are capable of simulating available bit rate, variable bit rate or constant bit rate traffic sources.

The switch accepts an incoming cell and switches it looking its switching table to determine which outgoing link should send it.

For variable bit rate (VBR) traffic generator the user sets the activation period, transmission rate, start time, link length and name of the text file where the log data will be saved.

For constant bit rate (CBR) and available bit rate traffic generator the user again sets transmission rate, start time, link length and name of the text file where the log data will be saved. But for ABR, the specified transmission rate is its ICR. In time this rate will be changed according to the network load.

The software contains an event manager which schedule and fire event. An event queue is maintained in which events are kept sorted by time. Whenever a new event is generated, it is placed in its proper order on the time scale. First event will be selected and depending on type of the event appropriate action will be performed. This event will then be removed from the list indicating that the event has been completed. Events can be scheduled at the current time or at any time in the future.

The simulator program includes a graphical user interface, which provides the user set one of the default network topologies and also the algorithm. Then the results are showed in the ACR, queue length and utilization graphs.

5.1 Parameter Settings

In our experiments, the following parameter values are used, unless specified otherwise:

- All links have a bandwidth of 155 Mbps.
- All VCs are unidirectional
- Traffic sources are greedy persistent sources, with infinite data to transmit.
- Start times and transmission rates of the sources are known. VBR sources are on/off sources.
- The source parameter rate increase factor (RIF) is set to one, to allow immediate use of the full explicit rate indicated in the returning RM cells at the source. Initial cell rate (ICR) is also set to almost peak cell rate, except when indicated.

- The switch target utilization parameter is set at 95% for ERICA algorithm.
- Queue length value at the switch output port is adjusted to track 300 cells for control algorithm.
- The switch measurement interval is set 1 ms or every count cells (e.g., count =50), whichever comes first.
- All configurations use 1 km links.
- CI and NI flags are not implemented.

5.2 Simulations

Although many Explicit-Rate flow control schemes have been proposed, there is not yet commonly accepted set of network configurations to compare and evaluate these schemes [24]. We describe here a collected set of network scenarios, drawn primarily from the research literature [4,24,25,26] that can be used for benchmarking ABR algorithms.

The algorithms presented in section 3 and 4 were examined with a set of different scenarios and the results were presented in the form of graphs for each configuration:

- Graph of allowed cell rate (ACR) in Mbps over time for each source
- Graph of ABR queue lengths in cells over time at each switch port
- Graph of link utilization (as a percentage) over time for each link

5.2.1 Three Source Configuration

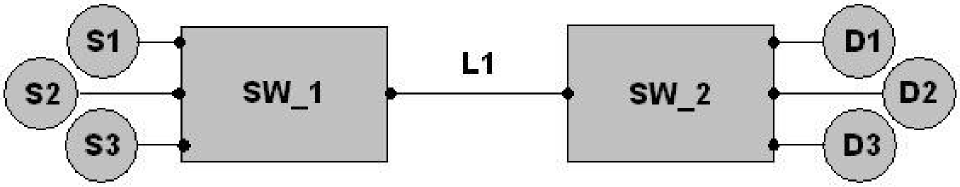

Figure 8 Three-source configuration

We used three-source configuration to test the fairness and effective use of ABR bandwidth. The configuration is shown in Figure 8. S1, S2 and S3 are greedy persistent sources and share a link as illustrated. They are active during the whole simulation period. Link bandwidth is 155 Mbs and ICR of sources is set to the link bandwidth.

The simulation results for ERICA on three-source configuration scenario are shown in Figure 9 and 10. Each source converges almost 1/3 x target utilization x link bandwidth, which is the max-min allocation of sources. Convergence is fast. There are no oscillations in the steady state. Since all sources start transmission at the same time, queue length at the switch output port suddenly grows up but queue size quickly returns its steady-state behavior.

ACR values of sources for control algorithm (Figure 11) for this configuration are quite similar to those for ERICA. ABR sources use their max-min allocation rates as soon as possible. As can be seen from Figure 12 queue length is regulated around a desired value of 300.

We have chosen link utilization factor of ERICA as 0.95 and 95% of the total bandwidth is used by ABR connections. Remaining 5% is used to drain the ABR queue when sustained congestion occurs. So utilization value of the link (Figure 10) oscillates between 95% and 100%. But in control algorithm the whole bandwidth is used (Figure 12).

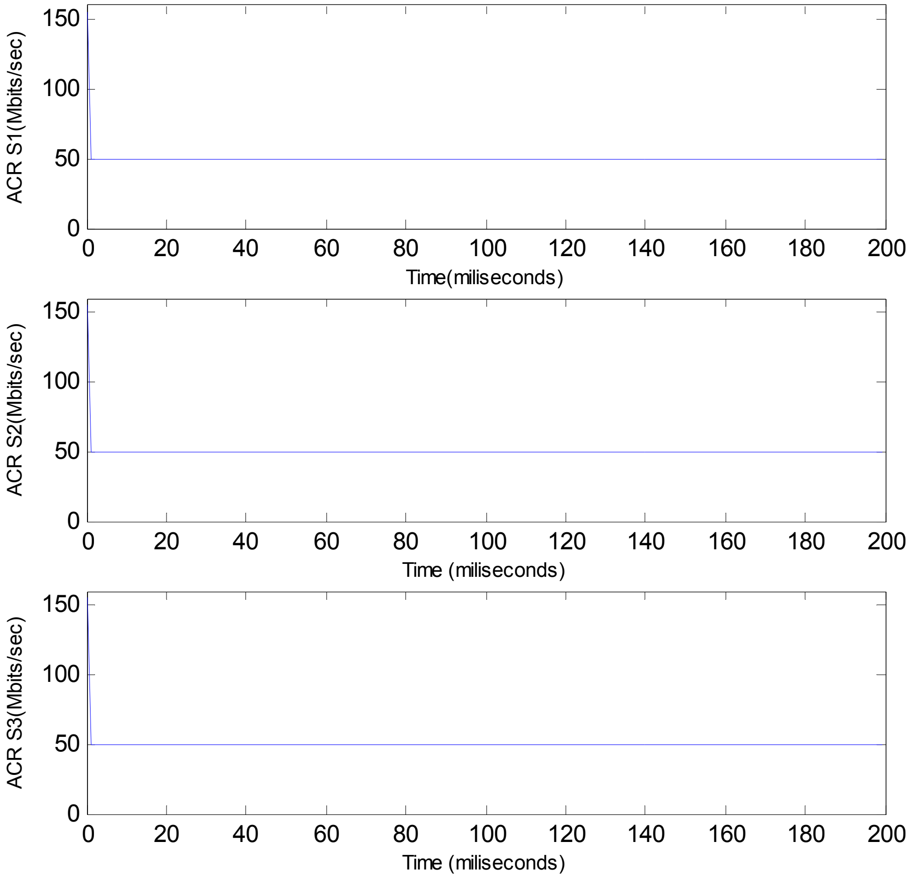

Figure 9 ACR values of three-source configuration for ERICA

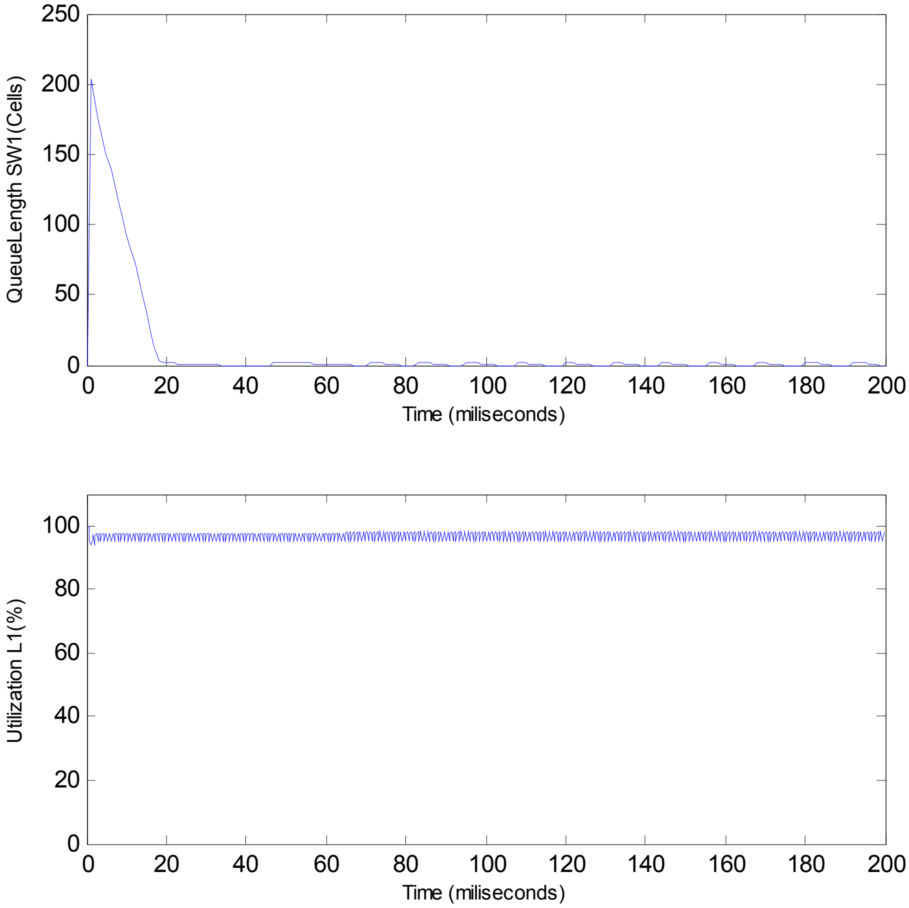

Figure 10 Queue length and utilization values of three-source configuration for ERICA

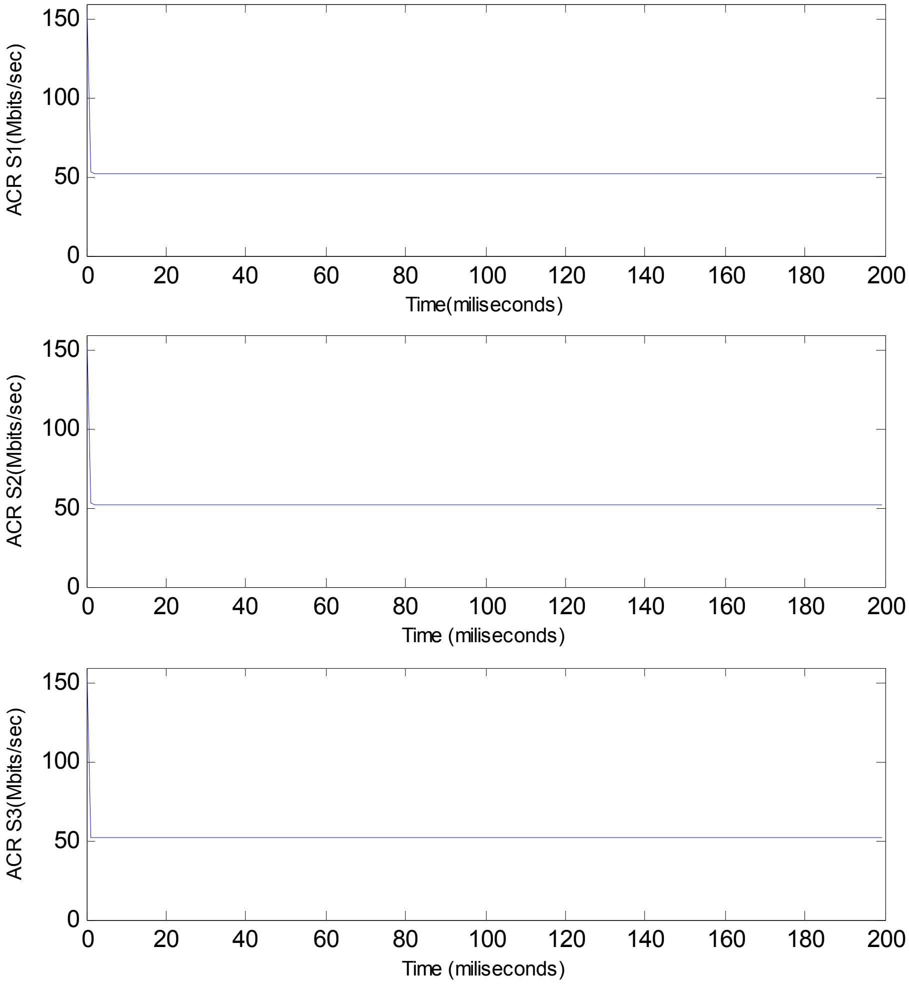

Figure 11 ACR values of three-source configuration for control algorithm

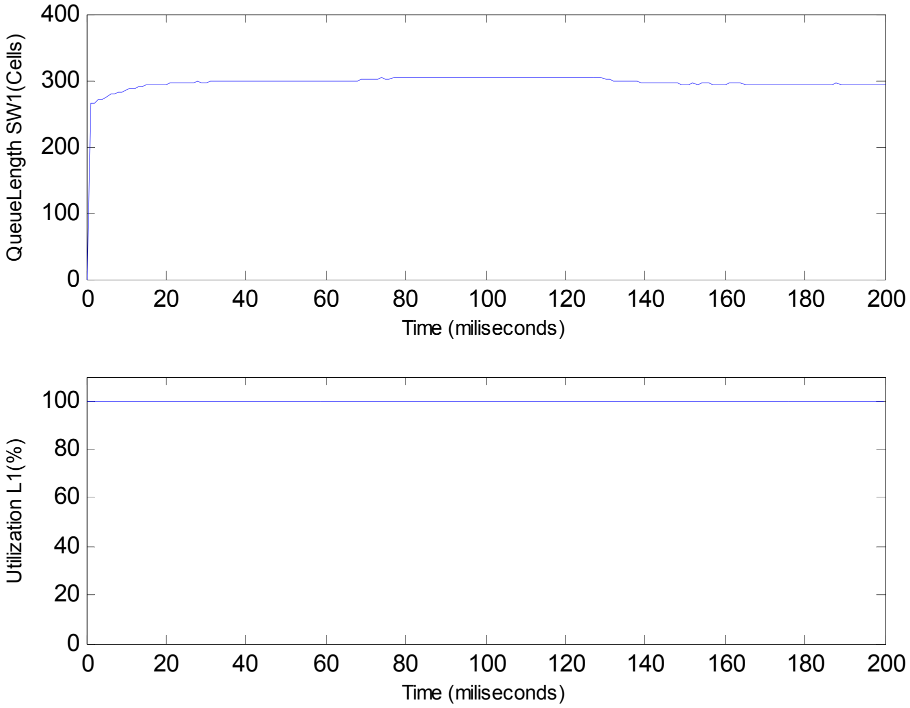

Figure 12 Queue length and utilization values of three-source configuration for control algorithm

5.2.2 Two Sources Staggered Configuration

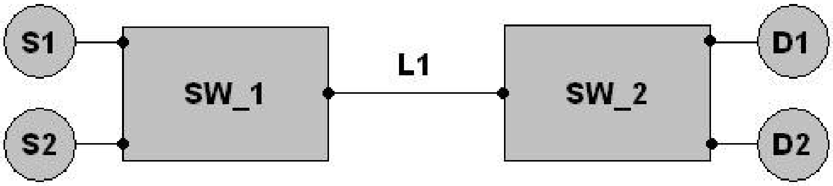

Figure 13 Two sources staggered configuration

To test the transient response, fairness and efficiency of the algorithm, we use a modified two-source configuration called "two source staggered configuration". The initial cell rate (ICR) for each source is 155 Mbps.

The simulation results for ERICA on this scenario are shown in Figure 14 and 15. In the first cycle of the simulation, only S1 is active and it uses whole link bandwidth. At t=10 ms, S2 starts transmission. After S2 started transmission S1 reduces its rate and share the link bandwidth with the second source.

Figure 15 shows that new arrival generates an impulse queue buildup, but then queue size returns its steady-state behavior.

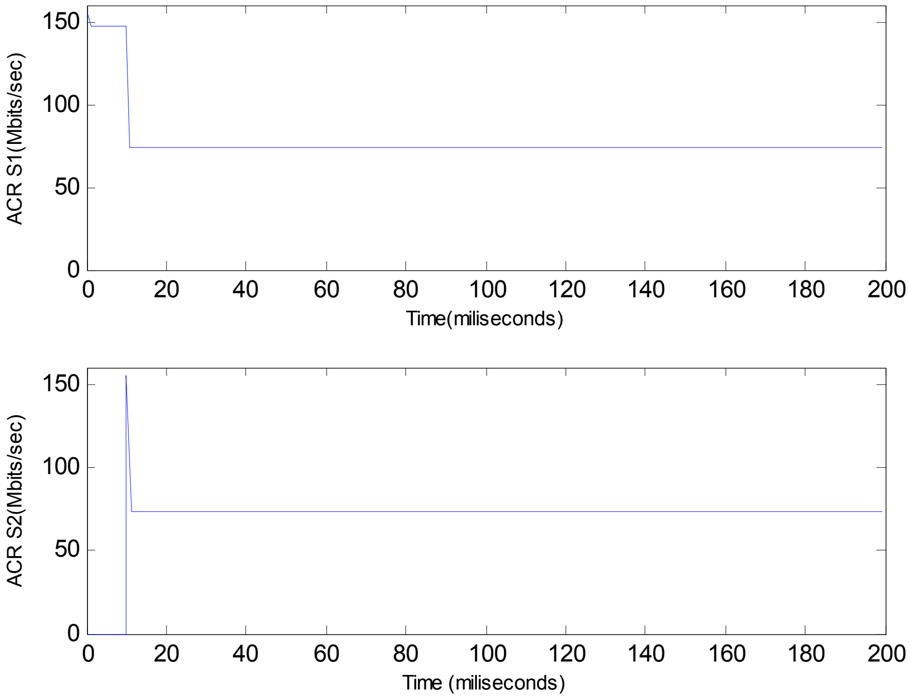

Figure 14 ACR values of two sources staggered configuration for ERICA

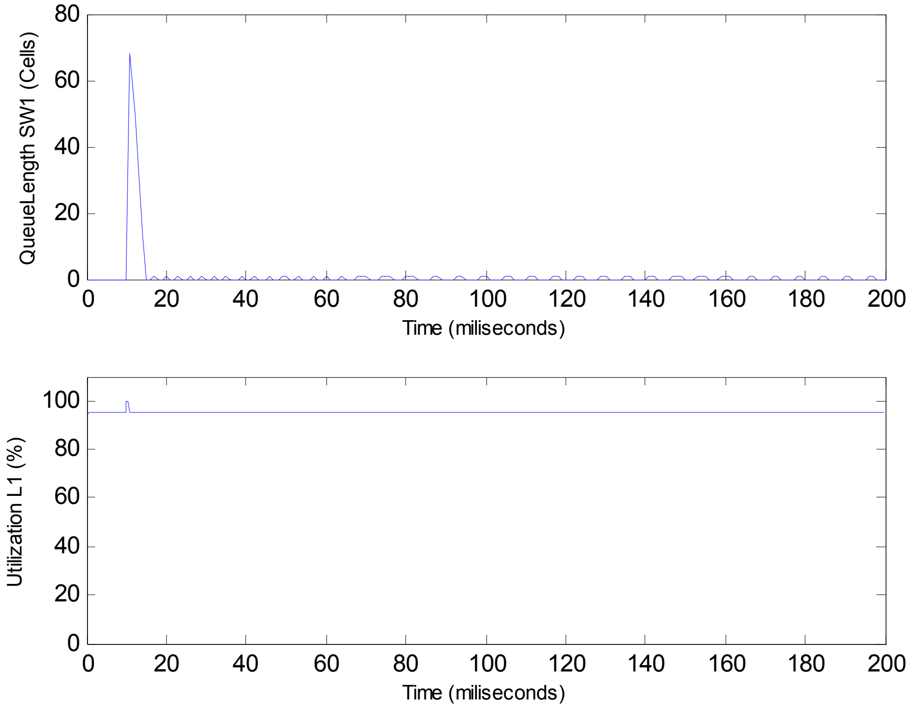

Figure 15 Queue length and utilization values of two sources staggered configuration for ERICA

The simulation results for control algorithm of this configuration are shown in Figure 16 and 17. Each source converges again its fair allocation. But this time convergence is not as fast as in ERICA algorithm. There are no oscillations in the steady state. From Figure 17 we see the full utilization of the link bandwidth. Queue length is very small in the first period of the simulation because there is only one source, which uses the whole link bandwidth at requested transmission rate. But immediately after second source enters in the network queue length starts to increase and tracks the desired value in steady state.

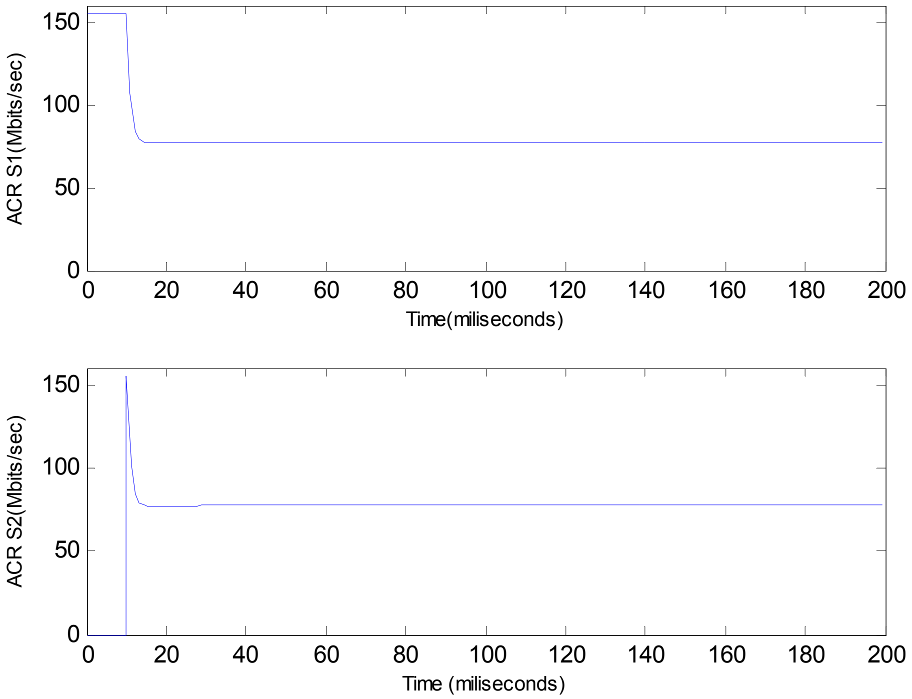

Figure 16 ACR values of two sources staggered configuration for control algorithm

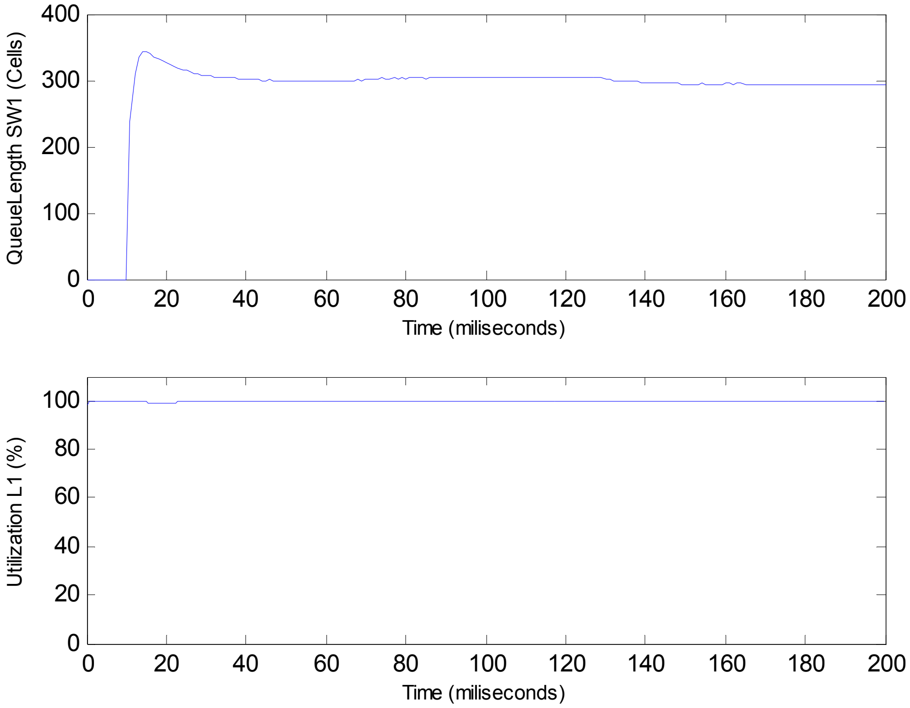

Figure 17 Queue length and utilization values of two sources staggered configuration for control algorithm

5.2.3 Three Source Configuration with VBR and CBR Background

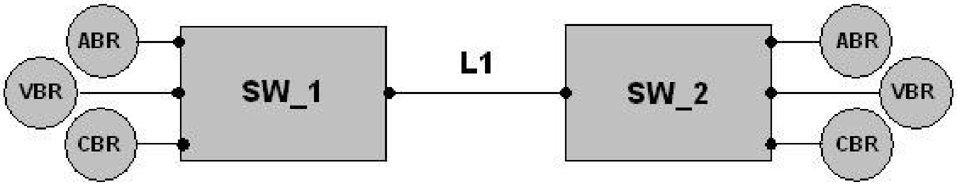

Figure 18 Three -source configuration with VBR and CBR background

Constant Bit Rate (CBR) and Variable Bit Rate (VBR) service classes have a higher priority than the ABR service. In cases of VBR and CBR traffic, the ABR capacity becomes a variable quantity [4].

Three-source configuration with VBR and CBR background is used to examine the behavior of the algorithm in the presence of VBR and CBR traffic. VBR source rate is 40 Mbps. It has 5 ms active and 5 ms inactive period. Peak cell rate of CBR is 30 Mbps. It will start at t=10 ms and will be active during the entire simulation period.

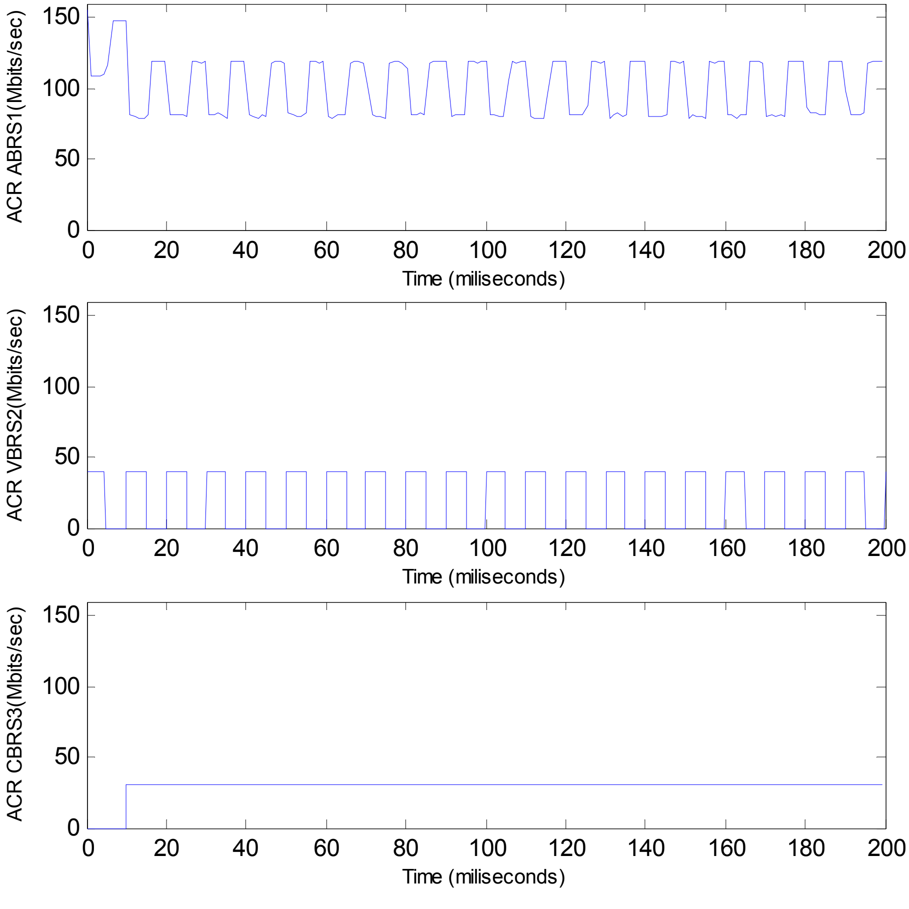

Figure 19 ACR values of three-source configuration with VBR and CBR background for ERICA

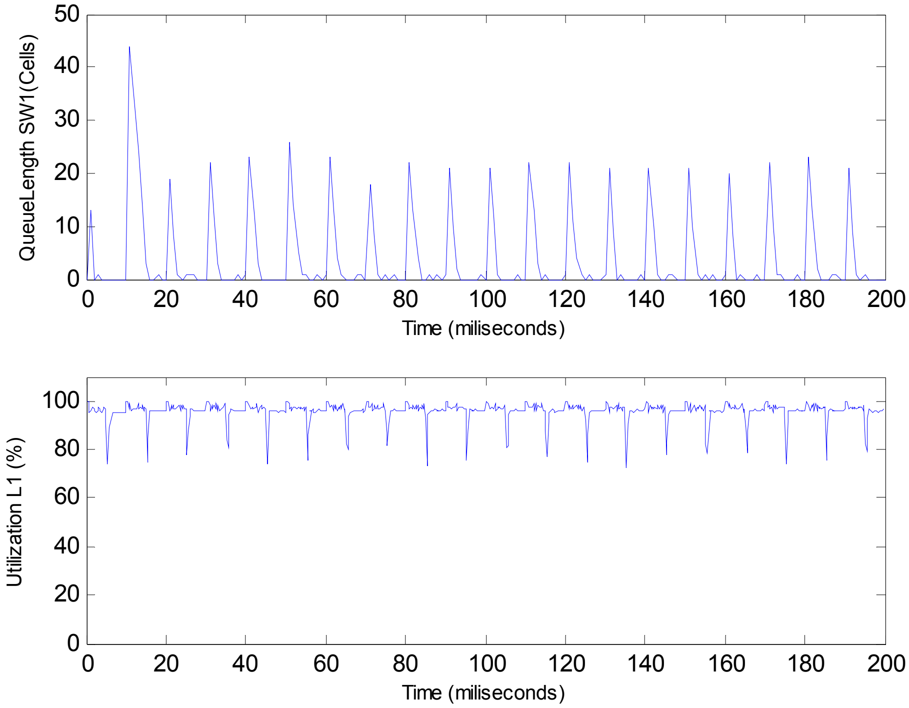

Figure 20 Queue length and utilization values of three-source configuration with VBR and CBR background for ERICA

In Figure 19, it is clear that ERICA rapidly detects the change in the available ABR capacity and gives the appropriate feedback to the sources. When the VBR source is active, the ABR sources rapidly reduce their rates. The total bandwidth that can be used by ABR is (link bandwidth–CBR–VBR). The utilization is generally high; the utilization drops reflect the time taken for the feedback to reach the sources: the feedback delay. The spikes in the queue length seen in Figures 20 also reflect the feedback delay, but the queue is rapidly drained.

The simulation results for control algorithm of this configuration are shown in Figure 21 and 22. As we said before, the CBR and VBR classes are assigned higher "priority" by the network switches and get a share of the link bandwidth first. So this algorithm also shows that the leftover capacity is used by the ABR source. Queuing behavior is shown in Figure 22. Queue length does not track a pre-specified value because there is only one ABR source. Its ACR continuously changes according to network conditions and on-off period of VBR source. By the time sources takes optimum rates queue is filled up with ABR cells.

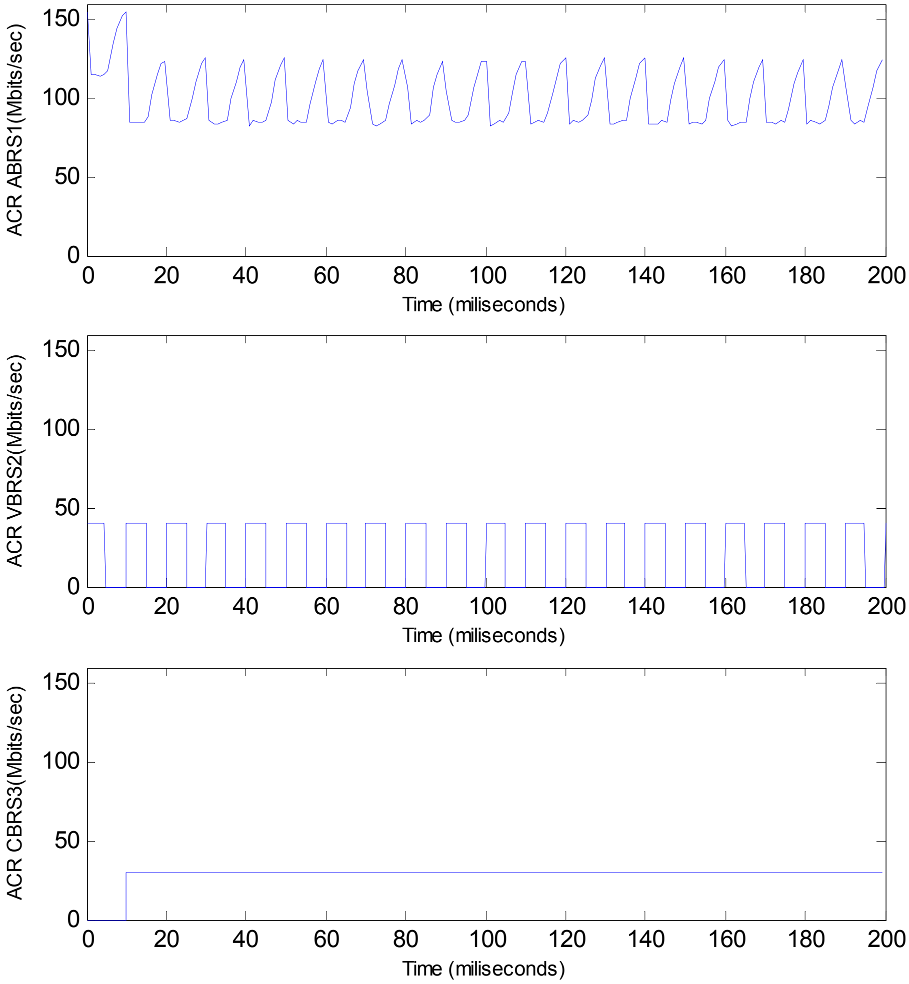

Figure 21 ACR values of three-source configuration with VBR and CBR background for control algorithm

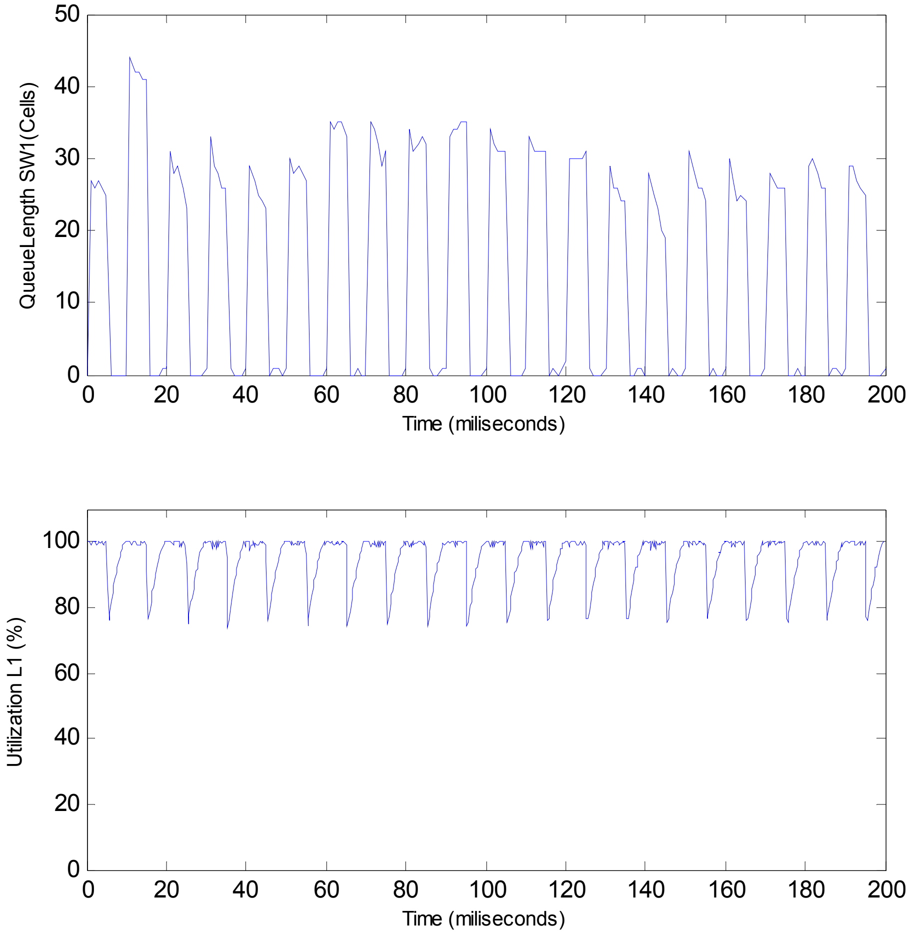

Figure 22 Queue length and utilization values of three-source configuration with VBR and CBR background for control algorithm

5.2.4 Upstream Configuration

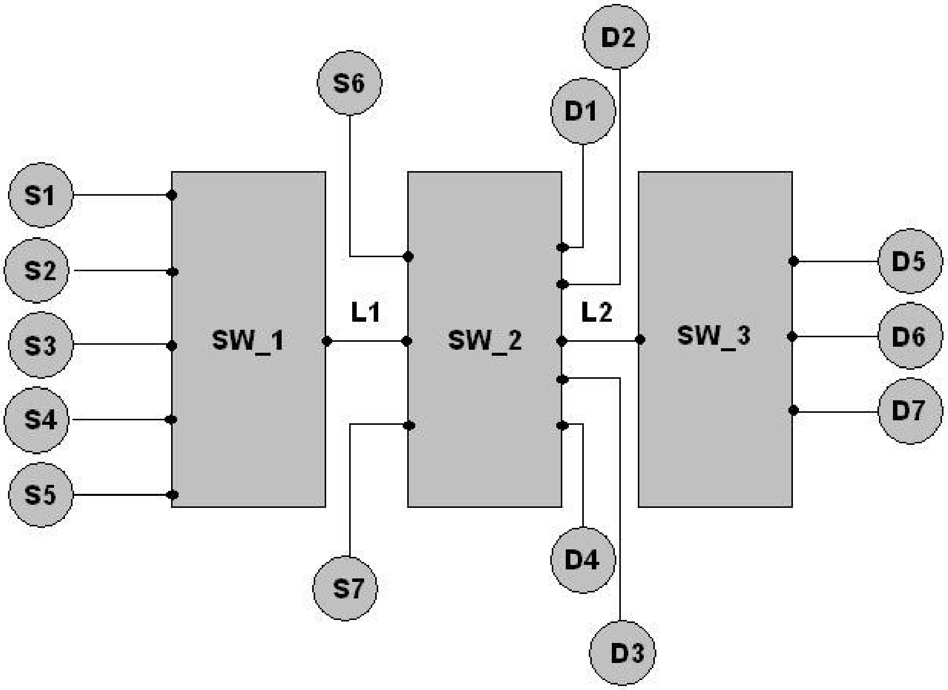

Figure 23 Upstream configuration

One of the configurations to study fairness of the scheme is upstream configuration as proposed in [4]. Our configuration consists of an upstream bottleneck.

The simulation results for ERICA using the upstream network scenario are shown in Figure 24 and Figure 25. The results show that all the sources receive their fair share bandwidth allocation. 5 sources (S1 to S5) receive approximately 30 Mbs. But S6 and S7 get around 60 Mbps ACR. The results are as we expected. At connection start up queue length shows a large transient.

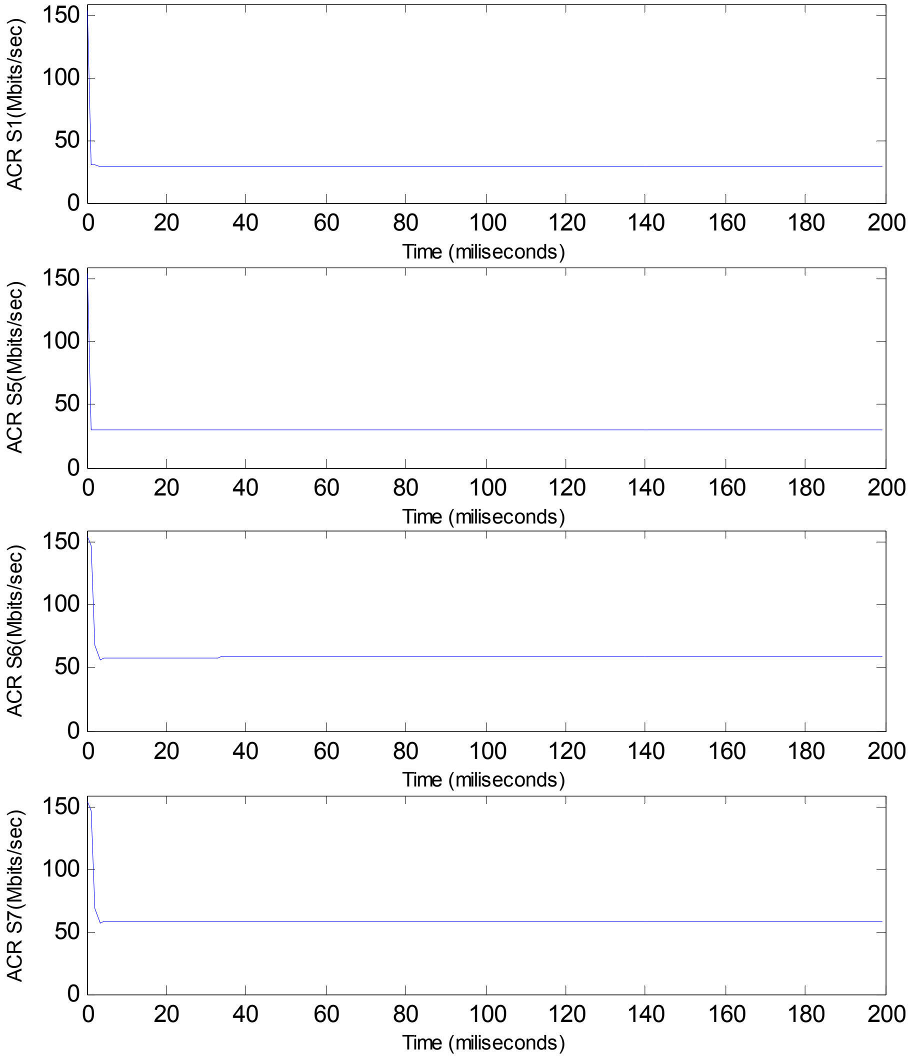

Figure 24 ACR values of upstream configuration for ERICA

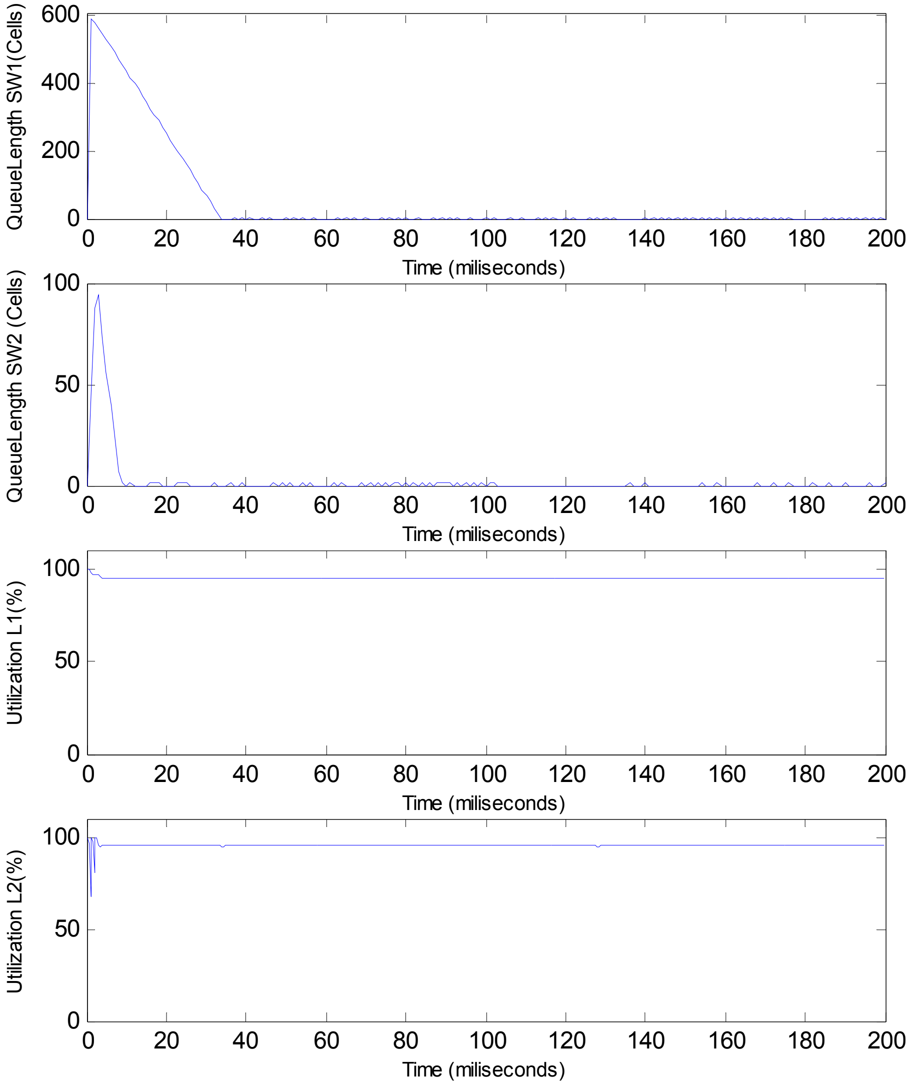

Figure 25 Queue length and utilization values of upstream configuration for ERICA

Figure 26 and 27 show the simulation results of control Algorithm. For this configuration we specified the desired queue size again as 300 cells. The convergence time of ERICA is slightly better. But also control algorithm performs well on this scenario. The sources can get their fair rates and queue length tracks the desired cell size. Except the initial underutilization of the link, utilization of two links is 100 %.

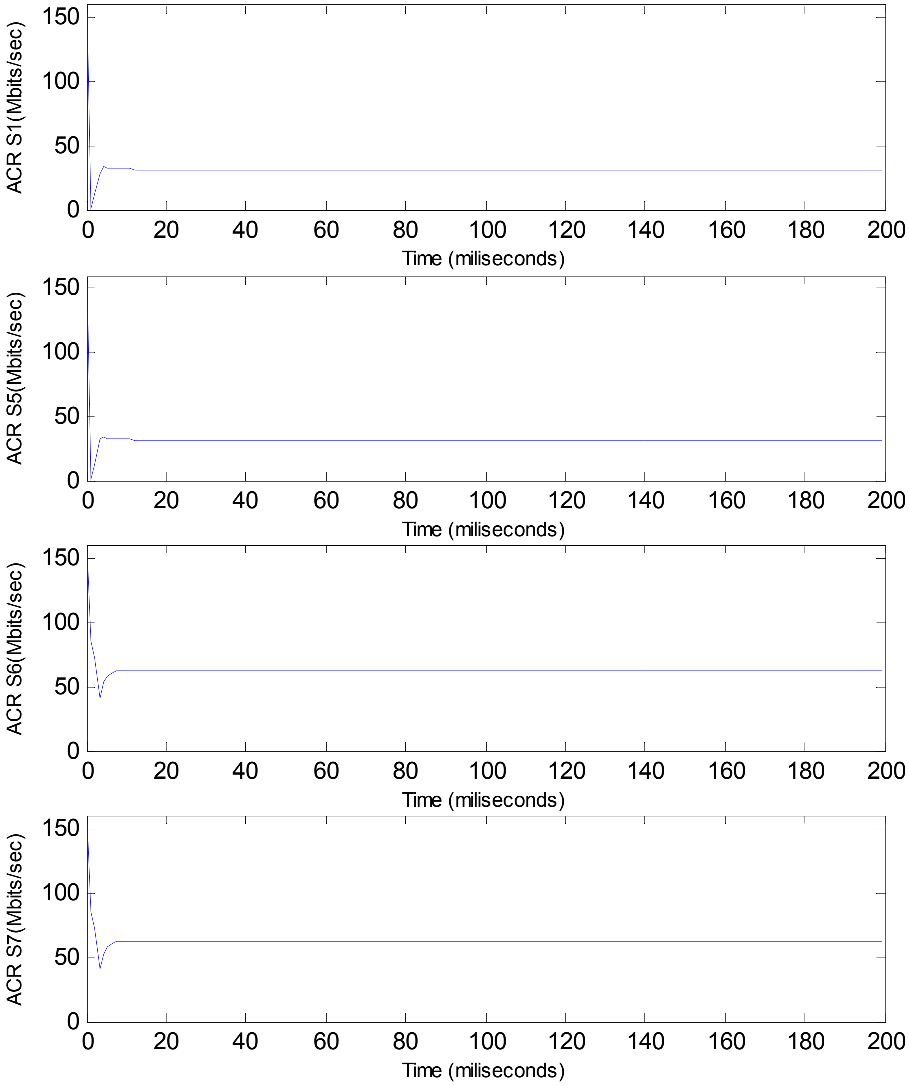

Figure 26 ACR values of upstream configuration for control algorithm

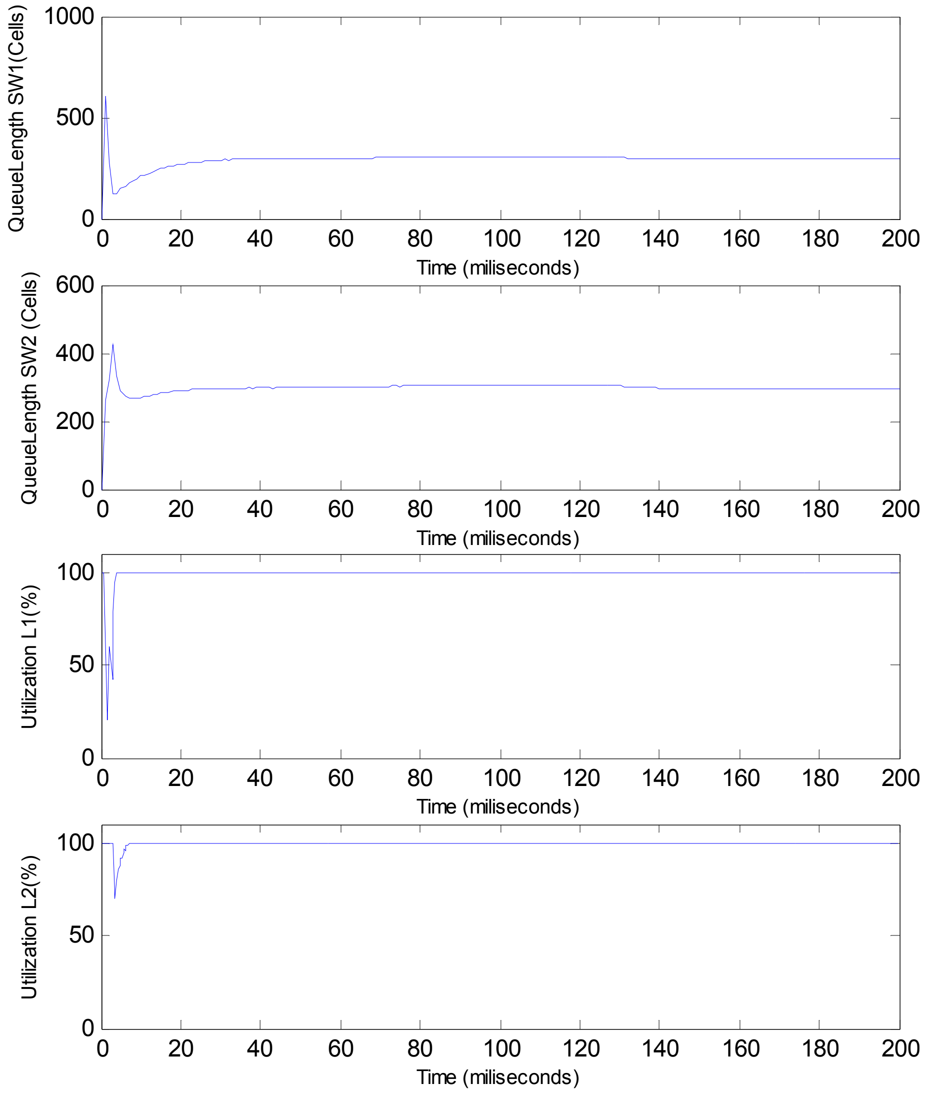

Figure 27 Queue length and utilization values of upstream configuration for control algorithm

5.2.5 Random Configuration

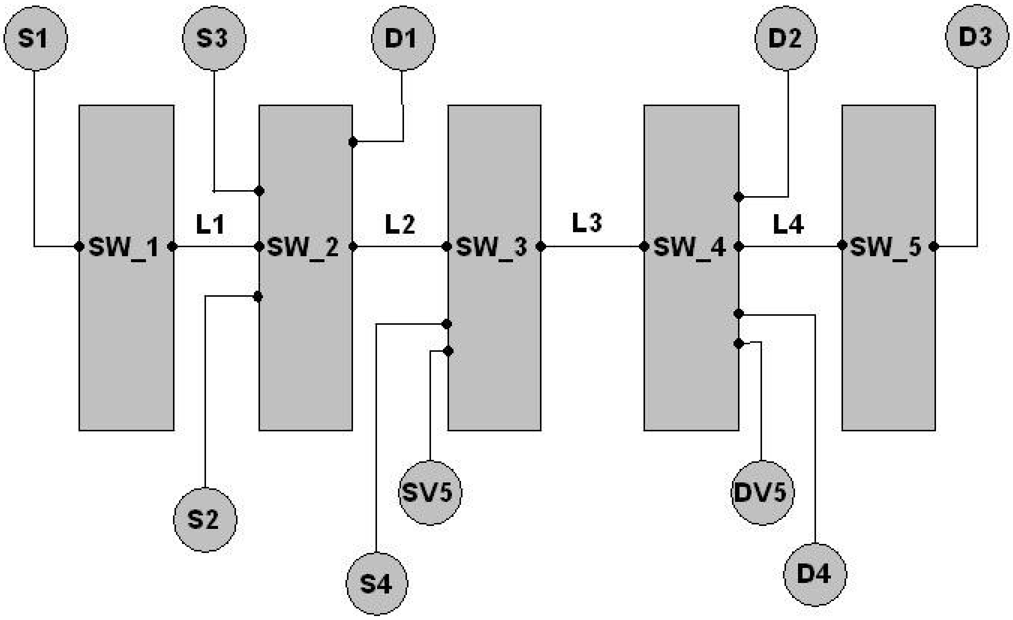

Figure 28 Random configuration

Finally, we have chosen a random configuration to test the operation of algorithms. In this scenario there are five connections. Four of the sources are ABR sources. The other one is VBR source with on-off period 30 ms. It will be active at t=30 ms and the transmission rate is 60Mbs. This is the scenario we model.

Results for ERICA are in Figure 29 and 30. As it is seen in Figure 29, source S1 uses the full bandwidth available because there is no other source, which shares the link L1 with S1. S2 and S3 share L2 so their allowed rate should be half of the link L2 but they also use L3 with S4 and SV5. Their optimum rates are calculated by switch SW_3. They cannot use their cell rate allowed by SW_2. The VBR background makes the ABR capacity variable at L3. The ACR’s of S2, S3 and S4 changes according to ON-OFF period of VBR source. During the OFF time, the switch experiences underload and may allocate high rates to sources. The duration of the OFF time determines how long such high rate feedback is given to sources.

As it is seen in Figure 30 at the beginning of VBR ON times ABR queues build up, since the switch is overloaded for a short time. Adjusted rates of sources to the new network condition may allow the queues to be completely drained before completed ON time. Utilization of L2 is high but it drops at the falling edge of VBR ON period to reflect the time taken for feedback to the sources. SW_2 is underloaded since S2 and S3 bottlenecked at L3 so utilization of L2 is less than 95 %.

The simulation results obtained using control algorithm (Figure 31 and 32) for this configuration are quite similar to those for ERICA. Compared to the performance of ERICA, this algorithm needs more time to converge of source rates to their optimum values. As can be seen from Figure 32 queue length at the output port of the third switch is regulated around a desired value of 300.

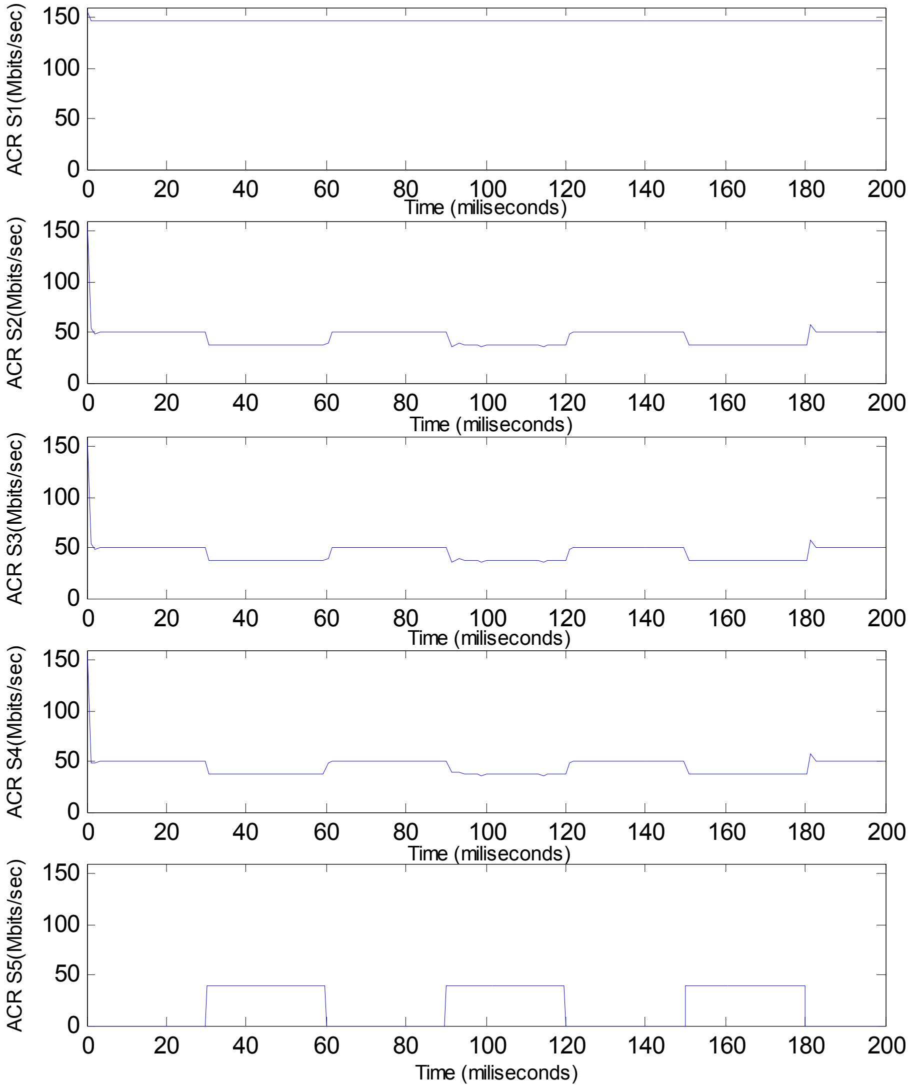

Figure 29 ACR values of random configuration for ERICA

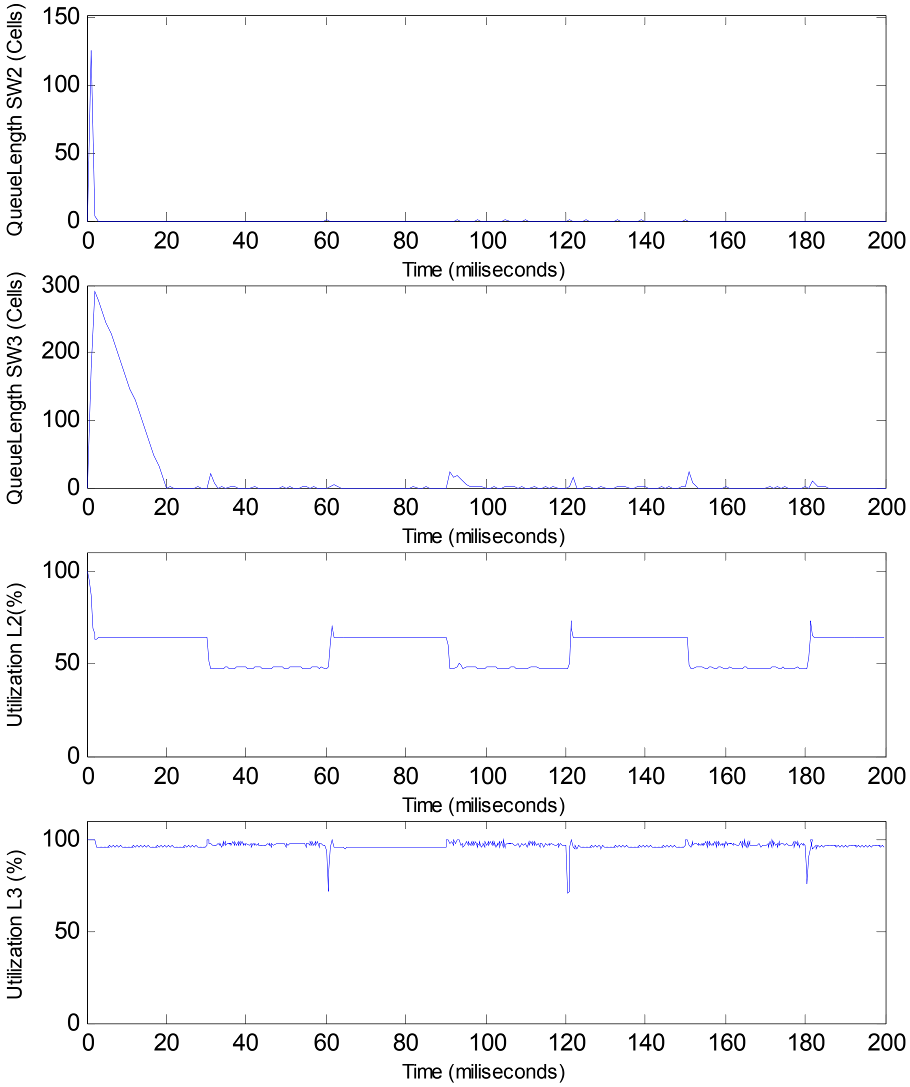

Figure 30 Queue length and utilization values of random configuration for ERICA

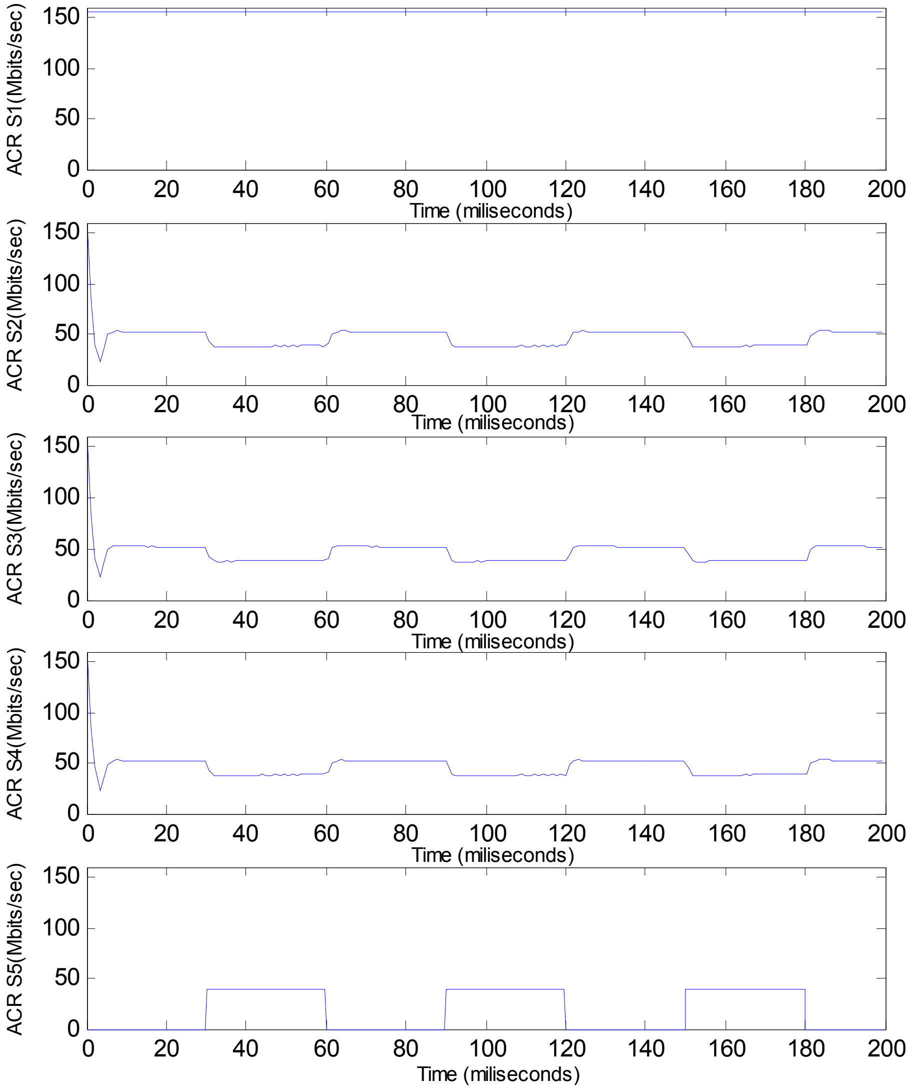

Figure 31 ACR values of random configuration for control algorithm

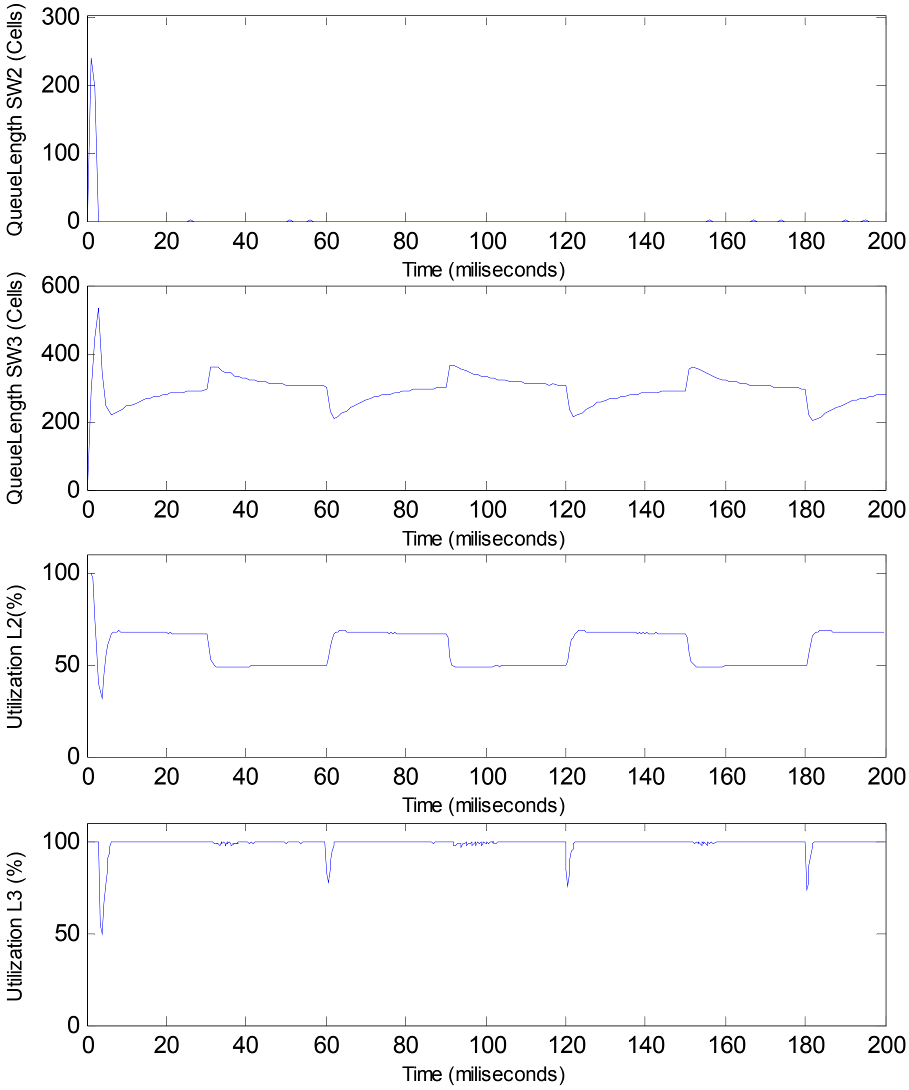

Figure 32 Queue length and utilization values of random configuration for control algorithm

CHAPTER 6

CONCLUSION

The objectives of traffic control and congestion control for ATM are to support a set of QoS parameters for all ATM services and protect the network from congestion in order to increase network performance. An additional role is to provide the efficient use of network resources.

Asynchronous Transfer Mode (ATM) is capable of supporting all classes of traffic (e.g., voice, video, and data) in one transmission medium. In ATM, traffic is divided five service classes to manage it easier. These service classes are categorized according to QoS requirements and traffic characteristics. The traffic management policy for them is different.

The available bit rate service class has been introduced to reliable support of data traffic that is loss-sensitive, not as delay sensitive as voice and video. Using feedback flow control, ABR sources adjust their transmission rates according to network condition. The main purpose of regulating the best-effort sources is to maximize the utilization of the total capacity available while avoiding congestion in the network.

In ATM networks, ABR sources adapt their rates to changing network conditions. This is achieved with explicit rate control messages, which are sent from switches to the sources using special cells called Resource Management cells. RM cells are generated by sources after every (Nrm-1) data cells are transmitted, where Nrm is generally taken to be 32. RM cells travel along the same path as the data cells and destination send them back through the network. On the path, each switch reduces the ER field of the RM cell to the maximum rate it can support, and sets CI or NI if necessary. When a source receives a BRM, it computes its ACR using its current ACR, the CI and NI flags and the ER field of the RM cell.

Early implementation of congestion control was based on binary feedback information where switches only mark a single congestion bit in the data cells in the event of congestion. The inherent shortcomings of binary approach has led to the consideration and implementation of sophisticated fair rate algorithms that compute fair rates for each connection. The current ABR congestion control specification gives considerable freedom to the switch vendors in designing and developing fair rate algorithms. A number of fair rate allocation algorithms have been proposed and studied in the literature. But almost all ABR flow control algorithms use the above basic RM cell framework. ER calculation algorithms differ according to calculation type of the optimum source rates.

The explicit rate indication for congestion avoidance (ERICA) algorithm was presented at the ATM Forum in February 1995. ERICA has been chosen as a comparison to the control algorithm because it is a well-known solution whose performance has been studied in many papers. The specification of ERICA was modified in order to propose a better solution (Jain; 1997). This latter version is not considered here.

Alternatively congestion control problem can be formulated where queue length is used to calculate explicit rate information. The mathematical model and a control based algorithm proposed in [19,27] provide fairness among all VCs with a minimal loss rate and maximal utilization of network resources by regulating the queue length at switches around a desirable level.

As stated before ABR traffic is loss-sensitive but not delay-sensitive. Hence one of the important features of explicit rate calculation algorithm for ABR service class is to avoid data loss. Tracking a nominal queue length is desirable in order to avoid losses due to overflow and waste of the buffer capacity due to underflow. Queue length information is not used in ERICA so it is difficult to control queue length to avoid buffer over flows. However, queue length remains bounded in an appropriate stochastic sense. In control method minimal data loss is achieved by regulating the queue length around a desirable level. This causes feedback delay and latency of transition response but as ABR is not delay sensitive we do not consider this effect of queue length control.

The switch averaging or measurement interval determines the accuracy of feedback. For ERICA this interval is used to measure the load level, link capacity and the number of active VCs for an outgoing link. Feedback is computed when a backward RM cell is received. In control algorithm this interval is used for the measurement of load level, queue length and also available link capacity. The length of the measurement interval establishes a tradeoff between accuracy and steady state performance. Shorter averaging intervals result in more feedback, but also more variation in feedback. Longer intervals impact the response time to load changes, but provide more stable feedback in the presence of asynchrony, and heterogeneous RTTs and FDs. In our simulations, control scheme's interval as well as ERICA scheme's interval is defined as every 0.001 seconds or 50 cells, whichever comes first.

In designing explicit rate calculation algorithms one of the main goals is to maximize link utilization. A parameter called 'target utilization' should be provided in ERICA. This determines the link utilization during steady state conditions. In ERICA excessively high values of target utilization are undesirable because they lead to long queues and packet loss, while low target utilization values lead to link underutilization. In our simulations we have used 0.95 as target utilization in order to reduce the ABR queue size and therefore, the cell delays.

In control algorithm there is no such a value to set in order to maximize link utilization. Avoiding losses due to overflow and wasting of the buffer capacity due to underflow is achieved by tracking a nominal queue length. If we choose a suitable queue length value to track, our network does not experience any queue overflow nor cell loss, while achieving 100% utilization. However ERICA cannot manage that with 100% target utilization. Especially during transient periods, and if the desired target utilization is set close to the full link rate, queue grows rapidly and results in heavy cell loss. Target utilization should be decreased to 95% to avoid queue overflows.

Another important criterion for the design and implementation of a rate allocation algorithm is fairness. Max-min fairness is commonly used notion for describing fairness. However, max-min fairness is ambiguous if ABR connections receive non zero bandwidth guarantees. In the case of nonzero MCRs, various fairness criteria exist. In our algorithms, the max-min fairness definition is considered. In simple terms, the goal of max-min allocation is to give equal shares of resource to all contending sources.

In order to achieve fairness, ERICA allows connections to increase their rates to a fair share during underload and fairness is achieved only after efficiency has been achieved, that is, the load factor is in the neighborhood of unity. Our simulation results illustrates that ERICA achieves the desired max-min allocation.

In ERICA the fair share is calculated as below:

$$Fair\ Share = \frac{T\arg et\ Utilization \times ABR\ Capacity}{Number of\ Active\ Connections}$$

VC share of each connection is also used to compute the optimum rates of sources. The VC share is calculated as:

$$VC\ Share = \frac{CCR}{Load\ Factor}$$

where CCR is current cell rate of each virtual circuit. In order to compute these quantities, the number of active virtual circuits and current cell rate value of every active virtual circuit should be measured in the average intervals. However, measurement can introduce variance in the system because of inaccuracies during measurement. These inaccuracies may cause unnecessary queues or rate fluctuations and limit the accuracy with which the goals are achieved. This is one of the limitations of ERICA algorithm.

But in control algorithm number of active sources should not be known to the switch. Fairness and queue length stability is achieved according to formula (4.11). Our simulation results have illustrated that the algorithm gives optimal allocations.

A fair rate allocation algorithm should take the minimum time to get close to the optimal rate. Sources should quickly reestablish the steady state behavior. The ability of providing rapid rate reductions under congestion is very important for the design and implementation of a fair rate allocation algorithm.

Two of the algorithms converge to the optimal max-min rate from any initial conditions. But convergence times of the algorithms are not similar. ERICA algorithm provides fast convergence to max-min rates and rapidly adapts to load and capacity changes. Control algorithm needs more time to converge to a steady state. The convergence time of the scheme is longer since it uses parameters whose values are chosen conservatively. In our simulations controller gains are picked up as (α, β)=(0.1, 0.01) in accordance with the theory developed. For a fixed value of α, the design parameter β can be chosen to tradeoff between the convergence and the magnitude of overshoots. A smaller value of β results in smaller overshoot, but a larger settling time. Both of the algorithms have an oscillation-free steady state performance.

The next requirement is simplicity of computations. For a fair rate allocation algorithm computations should be kept at minimum such that it can be implemented at a reasonable cost added to the switch design. The scheme should be simple to implement; it should not require measurements or logic, which are expensive. It should employ small number of variables. Further, the amount of memory required for the scheme should be minimal.

Control algorithm does not suffer from computational complexity. There are only two design parameters, namely α and β, to be tuned. To determine the explicit rate switch has to perform only four additions, two multiplications per output line. To perform these calculations switch needs to know queue length and available capacity at the end of averaging interval, and total number of incoming cells in this interval. The calculations do not require the number of connections going through the switch. So the memory to store required variables is not too much.

But ERICA needs a table of size n that account for the number of active connections. The complexity to manage this table can be O(n). The entire table must be reset every AI units of time.

As a result, ERICA algorithm is designed to achieve high link utilization with low delays and fast transient response. We presented performance analysis of the scheme using analytical arguments and simulation results. This scheme is also considered for implementation by several ATM switch manufacturers.

The control algorithm is presented to achieve max-min fairness and queue length stability under no centralized knowledge about the status of the network. This algorithm is easy to implement and adaptive to changing network conditions.

Design tradeoffs between performance objectives and implementation complexity for the two classes of explicit feedback control algorithms can be summarized in Table 3.

Table 3 Design tradeoffs for the two classes of explicit feedback algorithms

Type of Algorithm		Control Method	ERICA Algorithm
Performance Features	Convergence	Not as fast as ERICA	Fast
	Fairness	Achieves max-min fairness	Achieves max-min fairness
	Utilization of Sources	%100 Link Utilization during steady state	%95 Link Utilization during steady state
	Queue Length Control	Controls queue length	No queue length control
Implementation Characteristics	State Requirement	O(1)	O(n)
	Computational Complexity	O(1)	O(n)
	Buffering and Scheduling	One shared queue for every service class FIFO	One shared queue for every service class FIFO

To sum up, in this thesis, first the congestion control problem is introduced. Basic concepts of ATM networks and challenges of congestion control in ATM are given and then general traffic management functions are outlined. We described flow control framework of Available Bit Rate service category in details.

Then two congestion avoidance schemes for ABR service in ATM networks have been described. These two algorithms were simulated based on the ATM Forum's rate-based congestion control framework for ABR service. Using a set of benchmark network scenarios we have examined the efficiency, fairness, design and implementation aspects of the schemes. The transient and steady state performance and their adaptation to variable capacity and various source traffic models have also been described.

The rate allocation schemes discussed in these thesis supports point-to-point packet flow from one source to one destination. As a future work, with these algorithms flow control frameworks for ABR multipoint connections can be examined. We can also investigate effects of ABR source parameters on the congestion avoidance schemes to achieve the best performance.

REFERENCES

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

2. S. Kalyanaraman, "Traffic Management for the Available Bit Rate (ABR) Service in Asynchronous Transfer Mode (ATM) Networks", PhD Dissertation, The Ohio State University, 1997, xxiv + 429.

3. F.Lu, "ATM Congestion Control", URL: http://www.cse.ohio-state.edu/~jain/cis788-95/atm_cong/index.html.

4. S.Kalyanaraman, R.Jain, S.Fahmy, R.Goyal, and B.Vandolore, "The ERICA switch algorithm for ABR traffic management in ATM networks", IEEE/ACM Trans.Networking,vol 8, pp. 87-98, February 2000.

5. ATM Forum Traffic Management Specification Version 4.0, ftp://ftp.atmforum.com/pub/approved-specs/af-tm-0056.0000.pdf.

6. L.C. Alabern, "Traffic Management of the ABR. Service Category in ATM Networks", September 2004.

7. R. Jain, S. Kalyanaraman, S. Fahmy, R. Goyal, "Source Behavior for ATM ABR Traffic Management: An Explanation", IEEE Communications Magazine, November 1996.

8. The ATM Forum, "Traffic Management Specification Version 4.1", March 1999.

9. K. K. Ramakrishnan, P. P. Mishra and K. W. Fendick, "Examination of alternatives mechanisms for Use-it-or-lose-it", ATM Forum / 95-1999, December 1995.

10. R.Jain, "Congestion Control and Traffic Management in ATM Networks: Recent Advances and A Survey", Computer Networks and ISDN Systems, Vol. 28, No. 13, October 1996.

11. R. Jain, S. Kalyanaraman, R. Goyal, S.Fahmy, and R. Viswanathan, "ERICA switch algorithm: A complete description", ATM Forum/96-1172, August 1996.

12. A. Arulambalam, X.Chen, and N. Ansari, "Allocating Fair Rates for Available Bit Rate Service in ATM Networks", IEEE Communications Magazine, vol.34, no.11, pp. 92-100, November. 1996.

13. L. An, N. Ansari and A. Arulambalam, "TCP/IP Traffic over ATM Networks with ABR Flow and Congestion Control", Proc. 1997 IEEE Global Telecommunications Conf., Nov. 3-8, 1997, Phoenix, AZ., pp. 1845-1849.

14. E. Altman, T. Basar and R. Srikant, "Congestion control as a stochastic control problem with action delays", Automatica*,* 35(12):1937-1950, December 1999.

15. O. C. Imer, T. Basar, and R. Srikant, "A robust adaptive algorithm for ABR congestion control in ATM networks", Proc. IEEE ICCCN2000, pp. 48-53, Las Vegas, Nevada, October 2000.

16. O. C. Imer, S. Compans, T. Basar, and R. Srikant, "ABR congestion control in ATM networks", IEEE Control Systems Magazine*,* 21(1):38-56, February 2001.

17. O.C. Imer and T. Basar, "Optimal solution to a team problem with information delays: An application in flow control for communication networks", Proc. 38th IEEE Conference on Decision and Control*,* Phoenix, AZ, December 1999

18. T. Alpcan and T. Basar, "Global stability analysis of an end-to-end congestion control scheme for general topology networks with delay", IEEE Conference on Decision and Control ; Maui, Hawaii, pp. 1092-1097, 2003.

19. O. C. Imer and T. Basar, "Control of congestion in high-speed networks", European J. Control*,* 7:132-144, September 2001.

20. B.O.D. Anderson and J.B. Moore, "Optimal Control:Linear Quadratic Methods", Englewood Cliffs, NJ: Prentice-Hall, 1990.

21. Y. Afek, Y. Mansour, Z. Ostfeld, "Phantom: A simple and effective flow control scheme", in proceedings of the ACM SIGCOMM, August 1996.

22. N. Yin, M. G. Hluchyj, "On closed-loop rate control for ATM cell relay networks", in proceedings of the IEEE INFOCOM's 94, pages 99-108, June 1994.

23. http://www.atmforum.com/

24. Milan Zoranovic and Carey Williamson, "Performance and Robustness Testing of Explicit-Rate ABR Flow Control Schemes", in proceedings of the IEEE MASCOTS, College Park, MD, pp. 10-19, October 1999.

25. F.Chiussi, and Y.Wang, "An ABR Rate-Based Congestion Control Algorithm for ATM Switches with Per-VC Queuing", Proceedings of IEEE GLOBECOM'97, Phoenix, AZ, pp.771-778, November 1997.

26. F.Chiussi, Y.Xia and V.Kumar, "Dynamic Max Rate Control Algorithm for ABR Service in ATM Networks", Proceedings of IEEE GLOBECOM'96, London, UK, pp.2108-2117, November 1996.

27. O. C. Imer, T. Basar, and R. Srikant, "A distributed, globally-convergent algorithm for fair, queue-length-based congestion control", In Proc. 40th IEEE Conference on Decision and Control, Orlando, FL, 2001.

28. Y. Moret, S. Fdida, A. Fichou, C. Galand, "Evaluation of the ER algorithm ERAQLES in different ABR environments", 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Networks, Ilkley, West Yorkshire, U.K., 1997.