การวิเคราะห์สมรรถนะการควบคุมความคับคั่งสำหรับการให้บริการเอบีอาร์แบบหนึ่งจุดถึงหลายจุดในโครงข่ายเอทีเอ็ม

นาย นริศ รังษีนพมาศ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2544

ISBN 974-03-0374-9

PERFORMANCE ANALYSIS OF CONGESTION CONTROL
FOR POINT-TO-MULTIPOINT ABR SERVICE IN ATM NETWORKS

Mr. Naris Rangsinoppamas

A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy in Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
Chulalongkorn University
Academic year 2001
ISBN 974-03-0374-9

Thesis Title	Performance Analysis of Congestion Control for Point-to-Multipoint ABR Service in ATM Networks
By	Mr. Naris Rangsinoppamas
Field of Study	Electrical Engineering
Thesis Advisor	Professor Prasit Prapinmongkolkarn, D.Eng.
Thesis Co-advisor	Supot Tiarawut, D.Eng.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Doctor's Degree

……………………………………………Dean of Faculty of Engineering
(Professor Somsak Panyakeow, D.Eng.)

THESIS COMMITTEE

…………………………………………………… Chairman
(Associate Professor Somchai Jitapunkul, Dr.Ing.)

…………………………………………………… Thesis Advisor
(Professor Prasit Prapinmongkolkarn, D.Eng.)

…………………………………………………… Thesis Co-advisor
(Supot Tiarawut, D.Eng.)

…………………………………………………… Member
(Associate Professor Watit Benjapolakul, D.Eng.)

…………………………………………………… Member
(Assistant Professor Teerapat Sanguankotchakorn, D.Eng.)

นริศ รังษีนพมาศ,นาย: การวิเคราะห์สมรรถนะการควบคุมความคับคั่งสำหรับการให้บริการเอบีอาร์แบบหนึ่งจุดถึงหลายจุดในโครงข่ายเอทีเอ็ม (Performance Analysis of Congestion Control for Point-to-Multipoint ABR Service in ATM Networks) อ.ที่ปรึกษา: ศ.ดร. ประสิทธิ์ ประพิณมงคลการ, อ.ที่ปรึกษาร่วม : ดร. สุพจน์ เธียรวุฒิ, 114 หน้า. ISBN 974-03-0374-9

วิทยานิพนธ์ฉบับนี้ได้นำเสนอวิธีการปรับปรุงและวิเคราะห์สมรรถนะของการควบคุมความคับคั่งสำหรับการให้บริการเอบีอาร์แบบหนึ่งจุดถึงหลายจุดในโครงข่ายเอทีเอ็มโดยนำเสนอวิธีการรวบรวมข่าวสารคับคั่ง (Consolidation Algorithm) สองวิธีคือวิธี Rate-Queue Balanced (RQB) และวิธี Selective Backward Resource Management Feedback (SBF) โดยวิธี RQB นั้นใช้ค่าแบนด์วิดธ์ที่โครงข่ายสามารถรองรับได้และความยาวคิวของสวิตช์ในโครงข่ายเป็นพารามิเตอร์ในการกำหนดค่า Response-Accuracy Index (RAI) เพื่อควบคุมความรวดเร็วและความถูกต้องในการป้อนกลับข้อมูลไปยังแหล่งกำเนิดให้สอดคล้องกับสภาพการจราจรในโครงข่าย ในส่วนของวิธี SBF นั้นใช้วิธีการเฝ้าระวังและติดตามสาขาที่มีความคับคั่งมากที่สุดของโครงข่ายแล้วป้อนกลับแบนด์วิดธ์ของสาขานั้นกลับไปยังแหล่งกำเนิดแทนที่จะรอค่าจากทุกสาขาทำให้มีการตอบสนองที่รวดเร็วและข้อมูลที่ป้อนกลับเป็นข้อมูลที่ตรงกับสภาวะของโครงข่ายในขณะนั้น ซึ่งสามารถยืนยันได้ด้วยผลจากการทำการจำลองแบบ ในวิทยานิพนธ์ฉบับนี้ยังได้สร้างแบบจำลองทางคณิตศาสตร์เพื่อใช้เป็นเครื่องมือในการวิเคราะห์ประสิทธิภาพของโครงข่ายโดยแบบจำลองนี้สามารถประยุกต์ใช้ได้กับรูปแบบโครงข่ายที่หลากหลายและได้ผลใกล้เคียงกับการทำการจำลองแบบ นอกจากนี้ยังได้ศึกษาการทำงานร่วมกันระหว่างวิธีการรวบรวมข่าวสารความคับคั่งที่แตกต่างกันในโครงข่ายเดียวกัน (Interoperation) ในด้านความรวดเร็วในการตอบสนอง, การเกิดสัญญาณรบกวนจากการรวบรวมข่าวสารความคับคั่ง (Consolidation Noise) และผลกระทบเนื่องจากความไม่สมมาตรของเวลาประวิงครบรอบ (Round Trip Delay) พบว่าวิธีการรวบรวมข่าวสารความคับคั่งที่จุดแยกสาขา (Branch Point) ที่อยู่ใกล้แหล่งกำเนิดที่สุดจะมีความสำคัญที่สุดในการกำหนดสมรรถนะโดยรวมของโครงข่ายดังนั้นจึงควรพิจารณาใช้วิธีการรวบรวมข่าวสารความคับคั่งที่มีสมรรถนะสูงมีความรวดเร็วในการตอบสนอง, สัญญาณรบกวนจากการรวบรวมข่าวสารความคับคั่งต่ำและไม่มีผลกระทบเนื่องจากความไม่สมมาตรของเวลาประวิงครบรอบในขณะที่จุดแยกสาขาที่ไกลออกไปมีผลกระทบต่อสมรรถนะโดยรวมของโครงข่ายในระดับต่ำจึงควรพิจารณาเลือกใช้วิธีการรวบรวมข่าวสารความคับคั่งในด้านราคาที่ต่ำและสามารถสร้างได้ง่ายเป็นหลัก

ภาควิชา วิศวกรรมไฟฟ้า	ลายมือชื่อนิสิต..
สาขาวิชา สื่อสาร	ลายมือชื่ออาจารย์ที่ปรึกษา......................................
ปีการศึกษา 2544	ลายมือชื่ออาจารย์ที่ปรึกษาร่วม................................

3970763321: MAJOR ELECTRICAL ENGINEERING

KEY WORD: ASYNCHRONOUS TRANSFER MODE / TRAFFIC MANAGEMENT / AVAILABLE BIT RATE / POINT-TO-MULTIPOINT

NARIS RANGSINOPPAMAS, MR.: THESIS TITLE (PERFORMANCE ANALYSIS OF CONGESTION CONTROL FOR POINT-TO-MULTIPOINT ABR SERVICE IN ATM NETWORKS) THESIS ADVISOR: PROF. PRASIT PRAPINMONGKOLKARN, D.ENG., THESIS COADVISOR: SUPOT TIARAWUT, D.ENG., 114 pp. ISBN 974-03-0374-9

This dissertation investigates and improves the performance of the point-to-multipoint Available Bit Rate (ABR) service in ATM networks. We propose two new consolidation algorithms called “Rate-Queue Balanced” (RQB) and “Selective Backward Resource Management Feedback” (SBF). The RQB employs the Explicit Rate (ER) and Queue Length (QL) field in Backward Resource Management (BRM) cell as parameters to compute a Response-Accuracy Index (RAI) at a branch point. The RQB algorithm is adaptive to the network condition and the branch point uses it as an indication rather to work in a fast response or high accuracy way, however, at a little more expenses of buffer size in branch point nodes. SBF is an effective capacity tracking algorithm that can function in dynamic network scenarios. It achieves a fast response and low consolidation noise by selectively forwarding BRM cell from the most congested branch to the source instead of waiting for BRM cell from all branches. Simulation results show that the proposed algorithms outperform the already existed schemes in terms of response time and consolidation noise. The performance, i.e. the response time and Allowed Cell Rate (ACR) of the source is mathematically analyzed. They show relatively good agreement with the simulation results and can be applied for using in various network topologies. In addition, in this dissertation, the interoperability issue for the multicast ABR services is investigated. We address on a response time, a consolidation noise and an effect of asymmetrical Round Trip Delay (RTD) from branch point to destinations aspects. We found from the simulation results that the consolidation algorithm used at the most upper stream branch point (the nearest one to the source), especially in WAN configuration, plays an important role in determining the performance of the network. While consolidation algorithm used at the lower stream branch point affects the network performance insignificantly In LAN/MAN environment, due to a small difference in time delay, we can say that the consolidation algorithm insignificantly affects the network performance. Therefore, implementation simplicity of the consolidation algorithms should be a major consideration issue for employing to this kind of network.

Department	Electrical Engineering	Student's signature…………..……………..……
Field of study	Communications	Advisor's signature…………..………………..….
Academic year	2001	Co-advisor's signature…………………………..

Acknowledgements

Acknowledgements is the last page that I wrote in this thesis. I felt very happy when I was writing this acknowledgements. There are two reasons. Firstly, this thesis will be completed after finishing this page. I have written it for many months. Although, it is very difficult for me in writing this thesis because of my poor English but eventually, I am proud that I can do it. Secondly, I have a chance to express my gratefulness to many wonderful people who have helped me along the way in pursuit of this degree and made this a memorable experience.

The first person that I have to thank is my advisor, Prof. Dr. Prasit Prapinmongkolkarn. Without his guidance and support throughout the many years I did research under his supervision, it is impossible for me to finish my thesis. His encouragement, devotion and timeless advice are truly my inspiration. Besides knowledge, he also inspired me another good thing. He always reminds me that I have to discipline myself and be consistent in conducting research and he shows a good example. Unfortunately, I can do just a very few of his many good advices. Thank you so much with my deepest respect and sincere appreciation.

I would like to thank to the members of my Ph.D. committee and members of Electrical Engineering. Dept. staff. In particular, Dr. Somchai Jitapunkul and Dr. Watit Benjapolakul, Dr. Lunchakorn Wuttisittikulkij and Dr. Teerapat Sanguankotchakorn from Asian Institute of Technology for their comments and suggestions regarding my research work.

I would also like to thank to many of my colleagues for their help: thank to Khun Nattakij Phatcharatrisit for his sincere friendship and hardware supporting, to Khun Wisitsak Sa-niamsak and Khun Praphan Pavarangkul for their reviewing in software and simulation program. Especially, I would like to thank Khun Tanun Jaruvitayakovit for his insightful comments and good research companion. I have to thank all members of the EE. Dept. at King Mongkut's Institute of Technology North Bangkok, for providing me a warm place and facilities to do my research.

To my family, my parents and my sisters, I would like to express my heartfelt gratitude for all of their support, encouragement, cheerfulness and unwaivering belief in me. To my wife and my to be born soon daughter, many thanks to them for their understanding and their holiday time sacrificed for my study. I am lucky to have them.

I try my best to mention all people whom I am indebted. Please forgive me if I missed to express my gratefulness to some of you. However, this acknowledgements, though it is just a page of paper, it represents my deepest appreciation to all of you that have done an invaluable thing for me.

Contents

List of Tables

List of Figures

Abbreviation List

ATM	Asynchronous Transfer Mode
AAL	AIM Adaptation Layer
ABR	Available Bit Rate
ACR	Allowed Cell Rate
BECN	Backward Explicit Congestion Notification
BOM	Beginning of Message
BRM	Backward Resource Management
CAC	Call Admission Control
CBR	Constant Bit Rate
CCR	Current Cell Rate
CDV	Cell Delay Variation
CI	Congestion Indication
CLP	Cell Loss Priority
CLR	Cell Loss Ratio
CRC	Cyclic Redundancy Check
CTD	Cell Transfer Delay
DIR	Direction (of RM cell)
DES	Destination End System
EFCI	Explicit Forward Congestion Indication
ER	Explicit Rate
ERICA	Explicit Rate Indication for Congestion Avoidance
FEC	Forward Error Correction
FRM	Forward Resource Management
GCRA	Generic Cell Rate Algorithm
GFC	Generic Flow Control
HEC	Header Error Control
ICR	Initial Cell Rate
ITU	International Telecommunication Union
LAN	Local Area Network
MBS	Maximum Burst Size
MCR	Minimum Cell Rate
MID	Multiplexing 1Dentifier
Nrm	Number of data cells per RM cell
NI	No Increase

NNI	Network Node Interface
PCR	Peak Cell Rate
PDU	Protocol Data Unit
PNNI	Private Network Node Interface
PT	Payload Type
PVC	Permanent Virtual Channel
QL	Queue Length
QoS	Quality of Service
RDF	Rate Decrease Factor
RIF	Rate Increase Factor
RM	Resource Management
RQB	Rate-Queue Balanced
RTT	Round Trip Time
RTD	Round Trip Delay
SAR	Segmentation And Reassembly
SBF	Selective Backward Resource Management Feedback
SCR	Sustainable Cell Rate
SDU	Service Data Unit
SES	Source End System
SN	Sequence Number
SNP	Sequence Number Protection
SVC	Switched Virtual Channel
TBE	Transient Buffer Exposure
UBR	Unspecified Bit Rate
UNI	User Network Interface
UPC	Usage Parameter Control
VBR	Variable Bit Rate
VBR-rt	real time VBR
VBR-nrt	non-real time VBR
VC	Virtual Connection
VCC	Virtual Channel Connection
VCI	Virtual Channel Identifier
VPC	Virtual Path Connection
VPI	Virtual Path Identifier
VS/VD	Virtual Source/Virtual Destination
WAN	Wide Area Network
WFQ	Weighted Fair Queuing

CHAPTER 1

Introduction

In recent years there has been an explosive growth of the use of computer networks. As networks become more advanced and users expect more and more satisfactory and varied services, Asynchronous Transfer Mode (ATM) networks are one of the solutions of the broadband communication service for this ongoing requirement. ATM is being deployed widely in the information infrastructure. It is designed to support a wide range of traffic type for voice, data and video in a seamless manner. ATM networks operate on connection-oriented mode and transport data in fixed size of 53 byte-long packet called cells. A major advantage of ATM technology is the ability to provide a dynamic bandwidth allocation, which allows for the support of aforementioned traffic types. Nevertheless, such traffic types have a wide range of characteristics, e.g. bandwidth required, burstiness of the data, and loss and delay sensitivity. Furthermore, they may occasionally overutilize the network capacity. Therefore, the ability to take a preferential care to some network elements (sources, destinations or switches) is crucial. Congestion and flow control thus now play an important role [48], [49].

Congestion and flow control are complicated in the multipoint communication. In this dissertation, we focus the design of a congestion and flow control for the Available Bit Rate (ABR) ATM service to the point-to-multipoint connections. We investigate the problems of the point-to-multipoint consolidation algorithms. We analyze a mathematical model for estimating the transient response time and the Allowed Cell Rate (ACR) of the source. We also address the interoperability among various consolidation algorithms. The goal is to design an algorithm to efficiently use multipoint ABR communication and to develop a mathematical tool that can be used to systematically explain various phenomena in the network.

1.1 The Congestion and Flow Control Problems (Consideration issues) for Point to-Multipoint ABR Service

1.1.1 Problem Statement

In the congestion control of point-to-multipoint ABR service in ATM network, the most difficult challenges have been in providing congestion control algorithm which has a fast and accurate response from destinations and is fair for all network parties. In addition, a major challenge in developing this congestion control algorithm has been a lack of formal analysis. The interoperability among various consolidation algorithms is also under consideration. This dissertation will address these issues.

1.1.2 Problems in Point-to-Multipoint ABR Service in ATM Networks

In this section, the congestion and flow control problem for ABR traffic will be explained. We study to several survey papers for detailed descriptions of the various policies that have been proposed [24], [27], [30]and [57]. Congestion and flow control is a dynamic mechanism by which excess network resources are allowed to existing connections that can utilize them. ABR data connections can exploit excessive bandwidth and therefore efficiently use the resource in the ATM network.

We will assume that, at the time of connection setup, each ABR connection specifies a Minimum Cell Rate (MCR) and Peak Cell Rate (PCR) at which it wants to transmit its cells. We will also assume that, ABR flows cannot tolerate cell loss. At time *t*, the rate at which an ABR source is permitted to emit cells is called the allowed cell rate (ACR(*t*)). The ACR is also known as the allowed throughput. Under the above assumptions, MCR < ACR,(t) $\leq$ PCR for all *t*. No delay requirements are specified for the ABR service class.

1.1.2.1 Consolidation Noise

The consolidation noise is the fluctuation of the ACR of the source. It is generated by the branch point returning the BRM cells to the source before it has completely collected the congestion information from the downstream branches. The available rate received from all branches has to be minimized before sending back to the source. This can be done by using a per-branch calculation of the available rate. Nevertheless, previous algorithms use only one field that is updated every time a BRM cell is received and is reset to a PCR value every time the BRM is sent. Because of this simplification, some algorithms tend to feedback to the source with values that do not correspond to the most bottlenecked branch but to some other branches. This will result in an oscillation of the source rate and a consolidation noise is produced.

1.1.2.2 Transient Response

A response time of the consolidation algorithm in point to multipoint connection is the time duration counting from the source sending the first cell out until getting the first BRM cell back. In order to consolidate the bottleneck rate, some algorithms wait for the feedback from all branches to be received before sending a BRM cell to the source. This will incur a higher transient response delay.

1.1.2.3 RM Cell Delay

If the congestion arises at the switch at time t, the corresponding ABR sources of that multipoint session will not receive notice of this congestion until some later time, say $t + T$ where T is the amount of time taken by the RM cell from the congested switch to reach the source and is composed of propagation delays of the network links and a transient response delay.

Over the interval of time $[t, t + T]$, an active ABR source will transmit Ta cells where a is the source's ACR during this interval. Also, there are a certain number of ABR cells in transit when congestion arises. Consequently, a potentially enormous number of cells will be transmitted into, or are in transit within, an already congested virtual circuit; a significant number of these cells may be dropped due to buffer overflow in $[t, t + T]$. This phenomenon is a very significant obstacle for rate-based flow control schemes. If we would like to minimize the chance of buffer overflow we have to reduce the T value. The propagation delay is the physical value so it can not be removed, therefore, the transient response delay is the target to be tackled. The solution requires accurate tracking of the resource usage and the fast feedback of congestion information from the switch to the source.

1.1.2.4 Available Bandwidth Dynamics

The bandwidth usage in the network can fluctuate considerably due to the leave and join and change of the burst length of VBR users. Thus, ABR control schemes have to have these properties: a high enough sensitivity to track the resource usage and a fast enough response in order to properly allocates the left over bandwidth to the ABR sources. Care should be taken very much on designing the control scheme with the underlying properties. If the scheme is too slow or improper tracking to the bandwidth changing, then the throughput and network utilization is reduced. Similarly, if the scheme is too fast, it will result in oscillation of the source rate or consolidation noise. Hence, achieving the fast and accurate scheme is the designing challenge.

1.1.2.5 Fairness

The terms 'fairness' and 'fair share' are extensively used in ABR flow control. If the available bandwidth at a given link is enough to allocate to all of the users then the fairness issue will not arise. Bandwidth allocation among all users is necessary when the aggregate demand is higher than the available bandwidth. Generally, fairness dictates that users should be treated equally. Intuitively, bandwidth allocation scheme is fair if it does not offer a different treatment to users, either based on time order in which they make their requests, or on the particular location of their sources and destinations. ATM Forum's Traffic Management Committee has accepted the max-min fairness as a fairness calculation scheme. The max-min fairness suggests that a source with a bottlenecked connection at a given link will receive a bandwidth allocation which is at least as large as that of any other sources that also have a bottlenecked connection at the same link.

Fairness Definitions

Although there are many fairness definitions used in the resource sharing methodology, *max-min allocation* [10] fairness definition is the most widely accepted one. This is because its easy to implement feature and its allocation criterion are well suited to the ABR service in ATM networks. The definition in [10] is rather an abstract description. The more intuitive definition is given by S. Fahmy [58]. Below is an excerpt from the literature.

Definition: Max-min allocation: The max-min allocation vector is the feasible vector where the allocation of the sources with the minimum allocation is maximized. Given an allocation vector $\{x_1, x_2, \ldots , x_n\}$, the source that is getting the least allocation is, in some sense, the "unhappiest source". To achieve the allocations, find the feasible vectors that give the maximum allocation to this unhappiest source. Now remove this "unhappiest source" and reduce the problem to that of the remaining *n*-1 sources operating on a network with reduced link capacities. Again, find the unhappiest source among these *n*-1 sources, give that source the maximum allocation, and reduce the problem by one source. Repeat this process until all sources have been allocated the maximum that they can get. Intuitively, this means that all sources bottlenecked on the same link get equal rate, and if a source cannot utilize its fair share, the left over capacity is shared fairly among those who can use it.[58]

1.1.2.6 BRM/FRM Ratio

In point-to-multipoint connection, forward resource management (FRM) cells are replicated equal to the number of the branches. The number of backward resource management (BRM) cells may increase with this amount. Hence, in order to avoid the implosion of BRM cells at the source the BRM/FRM ratio which is the number of BRM cells received at the source for each cell sent by the same source is defined. The consolidation algorithm has to control this ratio to be one.

1.1.2.7 Scalability or Branch Point Level Sensitivity

There is an anticipation of ATM networks to cover a wide range of network sizes and configurations. Some previous proposed algorithms work well in certain network (e.g. LAN, single-level switch network) but poorly in others (e.g. WAN, multi-level switch) mainly due to the difference propagation delay. Therefore, the designed consolidation algorithm is necessary to have a scalable property. It should not function well only in the small and simple network but also work properly in a more complicated environment.

1.1.2.8 Complexity

ABR is notably more complex than the open looped control data services. Explicitly, the performance, in term of flow control function, is better for ABR. The complexity is inevitable because there is more control parameters and constraints. The challenge is how to design the control scheme to meet the pre-set requirement with the least complexity expense. There are a lot of quantitative values to measure the complexity. One of them is the hardware logic and the processing overhead. The hardware logic means the registers used in the processor unit, the buffer and parameter storage requirement. The processing overhead covers the control cell bandwidth and other needed control resources. These factors must be kept to the lowest because they reflect to an implementing cost.

1.2 Evaluation of the Consolidation Algorithm

To evaluate the efficiency of a consolidation algorithm, the advantages we get and efforts we put should be compared in various traffic conditions and for a wide range of network topology.

There are many advantages or benefits of flow control. One obvious benefit, from the user's point of view, is the increasing in average network resources utilization. In addition, the flow control may result in an increase in using an enormous left over bandwidth particular in a backbone ATM WAN configuration that comprises hundred megabits capacity. However, there are expenses for these advantages, additional switch and end systems hardware and computational power is required, the number of RM cells needed and any associated computational overhead and etc. As said before, a flow control scheme, or more specific, consolidation algorithm, is evaluated to be an effective one and worth adopting to use if the achieved performance exceeds the spent efforts both in reducing cost and complexity aspects.

1.3 Scope of the Dissertation

In order to improve the congestion and flow control scheme in point-to-multipoint ABR, in this dissertation, we study feedback control issue as it applies to the switch in ATM networks. The dissertation focuses on the designing of the consolidation algorithm to solve the problems in aforementioned aspects point-to-multipoint ABR services. Followings are the scopes and goals of this dissertation.

- To design and enhance the performance, in terms of response time, consolidation noise, link utilization and complexity of the congestion control algorithms which have been formerly proposed.
- To analyze the equations to mathematically illustrate the response time and allowed cell rate of the source for various consolidation algorithms including our proposed and other proposed ones.
- To investigate the interoperability of consolidation algorithms in ABR point-to-multipoint connection. We address on the response time and consolidation noise and the effect of asymmetrical round trip delay problems in various network environments.

1.4 Dissertation Organization

The main focus of this dissertation is on point-to-multipoint ABR service in ATM networks. The consolidation algorithms have been proposed. The performance evaluation method of the consolidation algorithms is mostly by simulation due to the complication of the network parameters. However, we did some parameters relaxation and analyzed a mathematical model to predict the allowed cell rate of the source. All the details are organized in the following manner.

In Chapter 2, the overview of the traffic management and services in ATM networks is given. Furthermore, the basic operation of closed loop congestion control and ABR flow control is also presented.

In Chapter 3, the major existing consolidation algorithms are given. Design criteria are set to improve the existing algorithms. Subsequently, two consolidation algorithms are proposed, namely, Rate-Queue Balanced (RQB) and Selective Backward Resource Management Feedback (SBF) in Chapter 4. The algorithms achieve the set criteria such as a fast response, low consolidation noise, fair bandwidth allocation and robust to the network dynamic by bandwidth tracking feature. This is one of the goals of this dissertation. Simulation results are illustrated with the broad range of network scenarios and comparison with existing algorithms are made.

In Chapter 5, another goal of the dissertation is presented. Analysis of response time and source rate of consolidation algorithms is analyzed. The derivation, even we have relaxed some parameters, is a bit complicated and need some effort to follow. However, it can be used as a tool to approximate the ACR of the ABR source during the time before all destinations get into the fair-shared state.

Interoperability of consolidation algorithms is investigated in Chapter 6. Again, we simulated all combinations of four algorithms in various network scenarios to see how the different consolidation algorithms using in different node effect the overall network performance. Conclusion of our work and future research are provided in Chapter 7.

In addition, there are two appendices. Appendix 1 shows some code of simulation program, RQB and SBF algorithm used in this dissertation and Appendix 2 is an excerpt from ATM Forum, AF-TM-0056.000 document, titled “Traffic Management Specification Version 4.0.

CHAPTER 2

Overview of ATM and Point-to-Multipoint ABR Service

2.1 B-ISDN Protocol Reference Model

To support multimedia applications, a network should offer a wide range of services over the network. Today services consume an enormous amount of bandwidth to ensure an acceptable quality. Multimedia comes from the merge of voice, video, data and image. The Integrated Services Digital Network (ISDN), adopted by Consultative Committee of International Telephone and Telegraph (CCITT), is set up to support a multimedia services in the same network and allows to have the integration of all services on one physical support. Broadband-ISDN (B-ISDN) which is the extension of ISDN and is designed to become the universal network based on the Asynchronous Transfer Mode (ATM) ATM has been accepted in a wide spectrum of telecommunications and data communications area. ATM was proposed to fulfil a fast packet switching concept that covers several alternatives, all with the same basic characteristics i.e. packet switching with minimal functionality in the networks. The properties that contribute ATM in qualifying as a fast packet switching network protocol are, for example, no link-by-link flow control or payload error control, using labels (Virtual Channel Identifier (VCI)/ Virtual Path Identifier (VPI)) instead of addresses in packets, using fixed and short 53 bytes size packet and switching in place of routing etc. Most of the data in multimedia services is compressed to reduce the cost and has a variable rate. ATM can use all available resources in the network as a "bandwidth on demand" so optimal statistical bandwidth sharing of the resources is obtained. Only one network needs to be designed, controlled and maintained for all technical scales that makes ATM to be termed as "one universal network".

The architecture of ATM is based on the Broadband-ISDN Protocol Reference Model documented in ITU-T Recommendation I.321[23]. The model consisting of a control, user, and management plane for three distinct layers: the physical layer, the ATM layer and the ATM adaptation layer (AAL). The physical layer defines how ATM cells are transmitted over a physical medium. The ATM layer is positioned above the physical layer and below the AAL and is responsible for flow control, cell header generation, multiplexing/demultiplexing of cells onto virtual connections, and VPI/VCI translation. The AAL sits on top of the ATM layer and maps higher level services, such as voice and data into the ATM layer. Figure 2.1 illustrates the B-ISDN Protocol Reference Model.

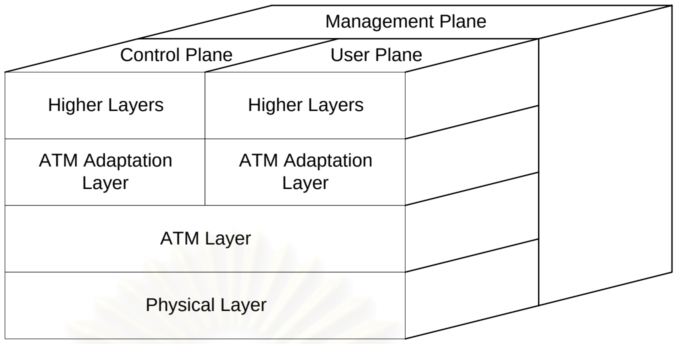


Figure 2.1 B-ISDN protocol reference model

The *user plane* (U-plane) provides for the transfer of user data. It contains a physical layer, ATM layer, and several AALs that support different higher-level services, such as as voice and data. The U-plane is the ATM protocol layer and function that supports application data transfer between two ATM end users.

The *control plane* (C-plane) provides for the signaling and control functions necessary to setting up and tear down connections. The C-plane supports an ATM SVCs with a signaling AAL and higher-level function. It shares the physical and ATM layers with the U-plane.

The *management plane* (M-plane) enables the U- and C-planes to work together.

2.2 Traffic Management in ATM Networks

The main goal of ATM traffic management is to ensure that the performance objectives can be satisfied for both new and existing connections. The mechanisms designed to achieve these goals are referred to as traffic management mechanisms. There are a number of traffic management techniques. Because ATM is designed to support integrated services, one technique is not appropriate for all. For example, one technique would be to allocate maximum bandwidth on a per-connection basis, this will suffer from a wasted bandwidth and does not optimize network resources. Another technique would be to provide a large number of buffers at the switches, at the cost of latency time, cost, and complexity. A third technique involves feedback of the network information to the source end systems in order to respond to changes in the available bandwidth by appropriately modifying their submission rates so that congestion is controlled or avoided.

Traffic management in an ATM network can be broken down into two areas:

- *Traffic control* is defined as the set of actions taken by the network to avoid congestion conditions.
- *Congestion control* as the set of actions taken by the network to minimize the spread and duration of congestion.

Most traffic and congestion control takes place at the network ingress or entry point to the ATM network. This prevents congestion by discarding or marking traffic that can lead to congestion situations. The chart in Figure 2.2 shows the possible traffic and congestion control schemes and the time frame for each scheme.

Traffic Control and Congestion Control Technique	**Time frame**
Traffic policing Traffic shaping Frame discard Buffer management	Cell / PDU input rate
Network feedback	Round-trip propagation delay
Connection Admission Control	Connection setup
Network Engineering / Design	Long term

Figure 2.2 Traffic management functions

2.2.1 Traffic Parameters

The traffic parameters defined are:

Peak cell rate (PCR. specifies the peak bandwidth that can be sent by the source into the network over a virtual connection.

Sustainable cell rate *(SCR)* specifies average data rate (in cell per second) over time that can be sent by the source to the network over a virtual connection.

Maximum burst size (MBS) specifies the number of cells that can be sent at the PCR rate.

*Minimum cell rate (MCR) s*pecifies the minimum bandwidth guarantee for an ABR service connection.

2.2.2 Quality of Service

QoS is a measurement on the delay and dependability that a particular connection will support. QoS is used by the CAC to allocate resources at connection setup time and by traffic management to ensure that the network performance objectives are met.

Peak-to-peak CDV specifies variation in cell transfer delay.

Maximum cell transfer delay (maxCTD) specifies end-to-end cell transfer delay.

Cell loss ratio (CLR) Lost cells/total transmitted cells.

Cell error ratio (CER) Errored cells/(successfully transmitted cells+errored cells).

Severely errored cell block ratio (SECBR) Severely errored cell blocks/total transmitted cell blocks.

Cell misinsertion rate (CMR.). Misinserted cells/time interval.

Further information on these parameters is available in the UNI 4.0 and Traffic Management 4.0 document from the ATM Forum.

2.3 ATM Service Architecture

The ATM Forum has defined a service architecture consisting of five ATM layer service categories that relate traffic and QoS parameters to network behavior. They are:

Constant bit rate (CBR)

Variable bit rate-real time (VBR-rt)

Variable bit rate-non-real time (VBR-nrt)

Available bit rate (ABR)

Unspecified bit rate (UBR)

2.3.1 CBR

CBR service is intended for real time applications that require tight constraints on CLR, CDV and CTD. The traffic contract is defined by PCR and by Cell Delay Variation Tolerance (CDVT). The CDVT defines the maximum cell delay variation for a stream entering the UNI, which should not cause cell rejection by the UPC. The source emits cells at a sustained PCR for the duration of the connection but could incur a wasteful in bandwidth and resources. Bandwidth renegotiations are possible under CBR. Examples of typical applications are CBR video and audio connections.

2.3.2 VBR-rt

VBR service is similar to CBR except that the traffic contract id defined by SCR in addition to PCR and CDVT. Cells can burst up to the PCR for a period of time but on average will be emitted at the SCR for the duration of the connection. The variation in the cell input rate enables multiple VBR-rt sources to be statistically multiplexed over the same physical connection to maximize network resources. Examples of typical applications are desktop video conferencing and voice.

2.3.3 VBR-nrt

VBR-nrt service specifies the same traffic parameters as the VBR-rt. It provides bandwidth guarantee at a PCR, but no guarantee for delay bounds. The variation in the cell input rate enables multiple VBR-nrt sources to be statistically multiplexed over the same physical connection to maximize network resources. Example of VBR-nrt application is airline reservation systems.

2.3.4 UBR

UBR service is intended for non-realtime applications, which do not have tight constraints on the cell delay and cell delay variation. In other words, UBR service is strictly for applications and connections that require no service guarantees. UBR sources can transmit up to link access speeds for however long that bandwidth is available or the "best-effort" fashion. UBR is a popular choice for running LAN applications over ATM because it is simple and mirrors LAN application behavior- requiring no priorknowledge on traffic rates or QoS, random transmissions at full available bit rates. The problem of using UBR is that there are no CLR guarantees for these applications, while many of these non-realtime applications expect a packet loss rate. This is one of the key motivations of the ABR service. Examples of UBR applications are file transfer and e-mail.

2.3.5 ABR

The ABR service is designed for non real-time applications which can control their transmission rate. The traffic contract is defined by PCR/CDVT and MCR. When the network has sufficient bandwidth, the connection is allowed to increase its cell rate up to Allowed Cell Rate (ACR): PCR ≥ ACR ≥ MCR. The value of ACR is updated periodically by a flow control algorithm in the transit ATM switches and delivered to the traffic source by the resource management (RM) cells. ABR applications are delay and loss tolerant. ABR is intended as the optimal ATM service for data networking applications because of the flow control and fair-access mechanisms. ABR flow control will be discussed in more detail in Section 2.4. Examples of ABR applications are LAN traffic and file transfer. Table 2.1 shows ATM layer services along with their specific attributes relating to traffic and QoS.

Table 2.1 ATM layer service parameters

Attribute	CBR	VBR-rt	VBR-nrt	UBR	ABR
Traffic parameters					
PCR and CDVT(PCR)	specified	specified	specified	specified*	Specified**
SCR, MBS, CDVT(SCR)	n/a	specified	specified	n/a	n/a
MCR	n/a	n/a	n/a	n/a	specified
QoS parameters					
Peak-peak CDV	specified	specified	unspecified	unspecified	unspecified
Max.CTD	specified	specified	unspecified	unspecified	unspecified**
CLR	specified	specified	specified	unspecified	*
Flow control					
Closed loop	unspecified	unspecified	unspecified	unspecified	specified

* Used either in CAC and UPC or information purposes only

** Represents maximum cell rate that ABR source may ever send. The actual maximum cell rate will be determined by network feedback.

*** CLR is low for ABR sources which adjust cell input rate according to feedback.

2.4 Closed-Loop Congestion Control

Because of the bursty nature of multimedia stream, it needs an efficient traffic management when transmitted over an ATM network. ATM networks have many advantages such as high trunk speed, flexible service type (bandwidth on demand) and high multiplexing capacity. The multiplexing of multimedia stream can reduce the burstiness of the aggregate traffic. However, the congestion may occur if the peaks of any application stream appears simultaneously. Congestion control and bandwidth allocation among the multimedia streams may resolve this problem and will improve the network resource utilization. In addition, flow control is used to regulate the traffic rate between the source and destination so that the network is not working in an overflow or underflow condition.

In a *closed loop,* the traffic source will adjust its cell input rate according to the feedback received from the network. The type of feedback received from the network can be binary (bits flipped to indicate congestion, increase/decrease rate) or explicit. For binary mode there are two notification techniques:

- *Forward. explicit congestion notification* (FECN) is known as Explicit Forward Congestion Indication (EFCI) marking, for which the source sends all user cells with set EFCI bit in the cell header to 0. If there is a congestion in a switch along the path, the switch will modify the EFCI to 1. Then, the destination will modify the forward RM cell to indicate CI=1 and returns them as a backward RM cells. An advantage of this FECN scheme is that it is compatible with the existing ATM switches with such EFCI functions, because they do not need to process RM cells. The processing of RM cells occurs only at the end systems for the EFCI marking scheme.

- *Backward explicit congestion notification* (BECN) is called relative rate marking. The congested switch either marks the backward RM cells with CI=1 or generate its own backward RM cell at the point of heavily loaded traffic to indicate the congestion. Based on that information, the traffic source will either increase or decrease the rate of input into the network. The BECN scheme reduces backward delay at the expense of increased complexity of the switch function that needs to process RM cells.

For explicit rate mode, ER introduces a function called *intelligent marking.* It factors in the current cell input rate at the source and an estimation calculated by each intermediate switch of the optimal bandwidth or each VC passing through the switch. Based on these two values, the RM cell returned to the source may contain a new explicit cell input cell rate. Fairness may be enforced because the switches can compute the maximum congestion free rate for an ABR source, based on the source VC's *current cell rate* (CCR) and available network capacity. This concept is illustrated in Figure 2.3

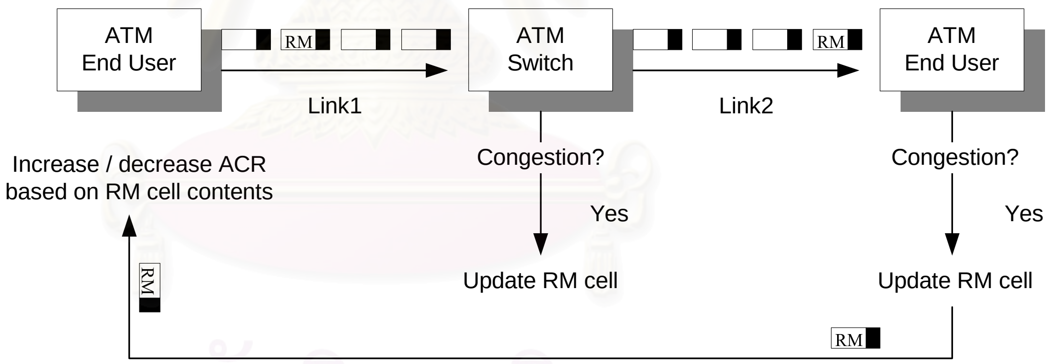


Figure 2.3 ER flow control

Resource management (RM) cells are used to communicate feedback information between ABR sources, destination, and switches. The RM cell contains fields that are marked or updated by intermediate switches as it is forwarded through the network. The RM cell is turned around by the ABR destination and returned to the source. The source, in turn, then adjusts its ACR based on the contents of the returned RM cell.

The switch must calculate the explicit rate (ER) of each ABR connection that passes through them by monitoring the load and detecting any congestion. Then the switch indicates the ER in the forward (or backward) RM cells. This introduces significantly complexity for the switch. The ABR explicit rate mode performance depends heavily on the switch algorithm used to calculate such explicit rates.

There are two techniques of closed loop congestion control for ABR service:

- Rate-based scheme that allows a source to adapt its specific cell input rate based on feedback from the network.

- Credit-based scheme that enables a sender to transmit cells to the consecutive receiver switch if there are available buffers (credits).

The rate-based scheme is better in WAN because the switches do not have to spare a large number of buffers due to a long propagation delay. This is backward compatible with older switches using EFCI. The credit-based scheme has an advantage on LAN side because there is completely no cell loss. A solution incorporating a choice of both schemes was rejected because it would require different techniques to be supported by vendor and the standards and would violate the concept of seamless ATM LAN/WAN integration.

2.5 ABR Flow Control

The ATM Forum has specified a service class, *available bit rate* (ABR), which most accurately reflects the behavior of LAN traffic. The main motivation for its development was the economical support of data traffic, where each packet of data is segmented into ATM cells, the loss of any one of which causes the re-transmission of the entire packet by a higher protocol layer. The ABR service would guarantee a particular cell loss ratio (CLR) for all traffic offered in proper response to network feedback. The basic parameters specified at ABR connection setup time are shown in Table 2.2

Table 2.2 ABR ER parameters [8]

Parameter	Description
Mandatory	
PCR	Peak cell rate -cell input rate source may never exceed
MCR	maximum cell rate -minimum cell input rate that source is guaranteed.
ICR	Initial cell rate -input rate at which source should send after idle period.
RIF	Rate increase factor-used to calculate increase in cell input rate upon receipt of RM cell, additive increase rate (AIR)=PCR*RIF.
RDF	Rate decrease factor-used to calculate decrease in cell input rate.
TBE	Transient butler exposure-negotiated number of cells that the source should send during start-up periods.
F'RTT	Fixed round-trip time-sum of the fixed and propagation delays from the source to the furthest destination and bark.
Optional	
Nrm	Maximum number of cells a source may send for each forward RM cell.
Trm	Provides an upper bound on the time between forward RM cells for an active source.
CDF	Cutoff decrease factor -controls the decrease in ACR associated with CRM.
ADTF	ACR decrease time factor -time permitted between sending RM cells before the rate is decreased to ICR.
Other	
ACR	Allowed cell rate -current cell input rate that a source is allowed to send.
CRM	Missing RM cell count-used to limit number of forward RM cells which may be sent in the absence of recieved backward RM cells.
TCR	Tagged cell rate-limits the rate at which a source may send out-of-rate forward RM cells.

2.6 Basic Operation

The source creates a connection with a call setup request. During this call setup, the values for a set of ABR-specific parameters are identified. Some values are requested by the source and possibly modified by the network e.g., the lower and upper bounds on the source rate, while others are directly chosen by the network e.g., the parameters characterizing the process for dynamically updating rates.

Once the connection is set , cell transmission begins. The rate at which an ABR source is allowed to schedule cells for transmission is denoted by ACR. The ACR is initially set to the Initial Cell Rate (ICR) and is always bounded between the Minimum Cell Rate (MCR) and the Peak Cell Rate (PCR). Transmission of data cells is preceded by the sending of an ABR Resource Management Cell (RM) cell. The source will continue to send RM cells, typically after every (Nrm-1, Nrm is equal to 32 by default) user cells transmitted and more frequently when its ACR is low. The source rate is controlled by the return of these RM cells, which are looped back by the destination.

The defined fields of the ABR RM cell include those listed in Table 2.3. The source places the rate at which it is allowed to transmit cells (its ACR) in the Current Cell Rate (CCR) field of the RM cell, and the rate at which it wishes to transmit cells (usually the PCR) in the Explicit Rate

(ER) field. The RM cell travels forward through the network, thus providing the switches in its path with the information in its content for their use in determining the allocation of bandwidth among ABR connections. Switches also may decide at this time to reduce the value of the explicit rate field ER, or set the Congestion Indication bit CI to 1. Switches supporting only the Explicit Forward Congestion Indication (EFCI) mechanism (by which an indicator in the header of each data cell is set under congestion) will ignore the content of the RM cell.

Table 2.3 RM cell

Field	Octet	Description
Header	1-5	Cell header with PTI='110'
ID	6	Protocol ID
DIR	7	Direction, 0--forward, 1=backward
BN	7	BECN, BN=1 indicates network or destination generated RM cell
CI	7	Congestion indication, CI=1 indicates congestion and cause source to decrease ACR
NI	7	No increase, used if switch detects impending congestion condition
RA	7	Request/acknowledge per 1.371, not used in ATM Forum ABR specification
Reserved	7	
ER	8-9	Explicit cell rate
CCR	10-11	Current cell rate, CCR--ACR when source generates RM cell
MCR	12-13	Minimum cell rate
QL	14-17	Queue length, not used in ATM Forum ABR specification
SN	18-21	Seq. number, not used in ATM Forum ABR specification
Reserved	22-51	
Reserved	52	
CRC-10	52-53	

When the cell arrives at the destination, the destination should change the direction bit in the RM cell and return the RM cell to the source. If the destination is congested and cannot support the rate in the ER field, the destination should then reduce ER to whatever rate it can support. If, when returning a RM cell, the destination had observed a set EFCI since the last RM cell was returned, then it should set the RM cell's CI bit to indicate congestion. As the RM cell travels backward through the network, each switch may examine the cell and determine if it can support the rate ER for this connection. If ER is too high, the switch should reduce it to the rate that it can support. No switch should increase the ER, since information from switches previously encountered by the RM cell then would be lost. The switches should try to modify the ER for only those connections for which it is a bottleneck, since this promotes a fair allocation of bandwidth. Also, switches should modify the ER content of the RM cells traveling on either their forward or backward journeys, but not on both.

When the RM cell arrives back at the source, the source should reset its rate, ACR, based on the information carried by the RM cell. If the congestion indication bit is not set (CI=0), then the source may increase its ACR by a fixed increment determined at call setup, towards (or up to) the

ER value returned, but never exceeding PCR. If the congestion indication bit is set (CI=1), then the source must decrease its ACR by an amount greater than or equal to a proportion of its current ACR, the size of which is also determined at call setup. If the ACR is still greater than the returned ER, the source must further decrease its ACR to the returned ER, although never below the MCR. A set NI bit tells the source to observe the CI and ER fields in the RM cell, but not to increase the ACR above its current value.

There are several congestion control schemes proposed for ABR traffic. Followings are some of the rate-based congestion control schemes:

Explicit Forward Congestion Indication (EFCI): A code-point in the header of ATM data cell as a single bit indicator of congestion in the forward direction of connection was used in the proposed scheme [25], [33], [34] a. During some intervals, the destination checks for the status of EFCI bit in the most recently received data cell whether it has been set or not. In case that it is not set, the destination transmits the RM cell containing permission to the source to increase its rate by fixed increment. If, the source does not get the permission to increase its rate over an interval of the same length, it decreases its rate by an amount proportional to its current rate. This scheme uses the concept of positive feedback that is a feedback is sent to increase the rate. This concept may seem consuming the bandwidth in normal network condition but in congested condition, it reduces an extra load of feedback messages on the network. This makes the scheme robust to lost or delayed feedback.

Proportional Rate Control Algorithm (PRCA): Like EFCI scheme, PRCA uses positive feedback, but limits bandwidth consumed by the ABR feedback to a fixed a proportion of total bandwidth available to ABR traffic [11]. Every Nrm cells (Nrm is set to 32 by default) forward cell, if this cell did not have an EFCI bit set, the destination generates one backward RM cell. The source increases its rate when it receives a backward RM cell. Otherwise, the rate decreases automatically. PRCA was found to have a fairness problem. Given the same level of congestion at all switches, the VCs travelling more hops have a higher probability of having EFCI bit set than those travelling smaller number of hops. This is so-called a ‘beat down’ problem.

Enhanced Proportional Rate Control Algorithm (EPRCA): The EPRCA source initialize ER field to their Peak Cell Rate (PCR) and set CI bit to 0. EPRCA calculates a rate as PRCA did by using the previous allowed rate and any single feedback received from the network, but then equates the new allowed rate with the minimum of this calculated rate and the most recent explicit rate received from the network. Each switch will then have the option of sending feedback using the explicit rate field, the congestion indicator or both [29]. The destination monitors the EFCI bit in data cells. If the last seen data had EFCI bit set, they set the CI bit to 1 in the RM cell. The switch can also set the CI bit in a returning RM cell if their queue length is more than a certain threshold. The sources respond by reducing rates after each cell by to a fixed a proportion of total bandwidth available to ABR traffic as in PRCA. The problem of EPRCA is a fairness problem. The switch congestion detection is based on queue length threshold therefore throughput was found to depend on how soon or how late you started. This problem can be fixed by using queue growth rate instead of queue length.

Explicit rate-based control scheme: The source generates the steady stream of cells called forward RM cells (each containing a field for explicit rate) and the destination loops them back call backward RM cell [50]. Each switch that the forward RM cell traverse across can then reduce the explicit rate in the RM cell if it is above the rate that the switch can support for the forward path. Switch will have the options of adjusting the explicit rate in the forward or backward direction of the connection. As Jain and Charny proposed, the RM cell also contains the rate that the source was allowed when it generated the forward RM cell [1]. This allows switches to allocate bandwidth fairly [7], [19]. This scheme adjusts the rate of a source more rapidly and with less oscillation than the single bit EFCI feedback. However, because switches have to calculate a fair share of bandwidth for each VC ([20], [21] and [56]), therefore, this scheme increases the switch complexity.

CHAPTER 3

Background of Congestion Control for Point-to-Multipoint ABR

3.1 Rate-Based Point-to-Multipoint Congestion Control and Consolidation Algorithms

In a rate-based point-to-multipoint ABR connection, the operation of feedback consolidation can be explained in Figure 3.1. A source transmits periodically the RM cells to the destinations or leaves. RM cells are then looped back by leaf node on each branch of multicast tree. Each RM cell contains a rate value agreed by all the nodes on the respective multicast branch. The branch point expects to receive RM cells within a certain time delay and their feedback values not exceeding a certain pre-agreed range and should not fluctuate due to the varying feedback received from different leaves. In addition, the consolidation algorithm at the branch point should avoid the feedback implosion problem, where the number of BRM cells received by the source is increasing proportional to the number of the destinations.

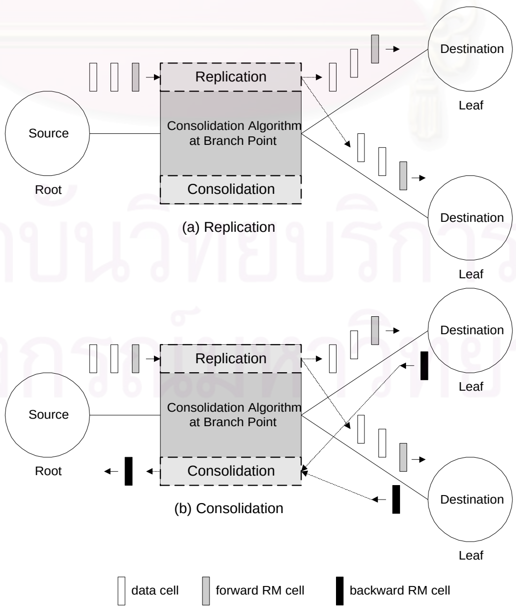


Figure 3.1 Function of branch point

3.2 Problems in Consolidation Algorithm

In ATM point-to-multipoint connections, the consolidation algorithm at the branch point is used to consolidate the BRM cells from the branches. The preferred characteristic of the consolidation algorithm is the quickness of the responding to the network change and the accuracy of the congestion information feeding back to the source. For some previously proposed algorithms, the branch point has to wait for the BRM cells from all branches and send the BRM that has the least ER value to the source. This will lead to an under utilization of the link and a slow response of the branch point in case that some branch is not a responsive branch. Some literatures proposed the algorithm that is not wait for all BRM from all branches. These algorithms may result a fast response but due to lacking of a completion of congestion information they may lead to an oscillation of the source rate or consolidation noise in the network. Hence, the challenge of designing a consolidation algorithm ABR point-to-multipoint communications is how to meet the two different requirements, i.e. fast and accuracy, with the least expenses of cost and complexity.

3.3 Consolidation Algorithm Design Criteria

The Traffic Management Specification version 4.1 [8] provides a guideline as a basic framework for a complete specification of point-to-multipoint ABR service. The defining characteristics are the dynamic allocation of available bandwidth (Rate allocation algorithm) and the flow-controlled transport of data from the root to each responding leaf (Consolidation algorithm). However, these characteristics are functionally defined, but their operating policies are implementation specific. The major concern at a branch point is how to consolidate the BRM cells from destinations to the source. The branch point that waits for the BRM cells from all destinations in the multicast tree will introduce a slow response or a consolidation delay. On the other hand, if the branch point consolidates some or even a single BRM cell from the destinations and then returns to the source, consolidation noise is introduced.

There are many criteria to design the consolidation algorithm. Which criteria to be used to meet the main requirements i.e. fast response time and high accuracy are listed as follows:

- Point of BRM cell feedback
- Waiting for BRM cell from all branch
- BRM to FRM ratio
- Scalability
- Complexity

- Interoperability

There were several papers proposing the consolidation algorithm ,[30],[35], [57] and [59]. The consolidation algorithm can be categorized into two types, namely, the slow response with low noise and the fast response with high noise. In this thesis, we proposed a new efficient consolidation algorithm. The algorithm offers a faster response, less consolidation noise, better link utilization and less complexity. The details of the algorithm will be explained later. There were many papers [2]-[4], [6], [17],[18], [36]-[38], [41] and [41] proposing a performance analysis of the unicast ABR services. For ABR multicast, there are some papers which proposed the consolidation algorithms but only a few literatures analyzed the algorithm mathematically [22],[60]. We analyzed and compared the performance of various types of consolidation algorithms. The response time, source rate and queue developed at the branch point were investigated. Many ATM vendors have already deployed switches with ER rate allocation capabilities and it is likely that different vendors will implement different techniques of ER rate allocation algorithm. For consolidation algorithm, we are aware that there is a possibility that the branch points (switch) will adopt different algorithms. This brings up the issue of interoperability between different consolidation algorithms in the network that will be detailed in later chapter. To our best knowledge, there has been so far no paper addressing the interoperability issue in point-to-multipoint ABR services ever been presented. Hence, we also investigate the interoperability issue and the impact of using different types of consolidation algorithm here in this dissertation.

3.4 Review of Previous Work

The traffic management problem for point-to-multipoint connections is an extension to the traffic management for unicast connections. There are many problems arise, especially, the consolidation of the feedback information from different destinations to a branch point. There are some literatures proposing a framework to cope with this problem[22], [24], [57] and [59]. The common goal in the early phase of the research in this topic is to ensure that all destinations receive all cells from the source. This requires that the source should be controlled to the minimum rate supported by all destinations. The minimum rate is the technique most compatible with the typical data requirement where no data should be lost. To meet this goal, it seems not too difficult but this may suffer from an unacceptable delay and low network utilization. The newly proposed papers including this thesis try to improve these drawbacks with the expense of some additional buffer at the switch element.

Although several consolidation algorithms were proposed previously, we choose four representative algorithms. The four algorithms used in this thesis are the wait-for-all consolidation algorithm [60], the not wait-for-all consolidation algorithm [22], Fahmy algorithm [57] and our proposed, SBF algorithm. The idea behind choosing these algorithms is to illustrate how the response time and consolidation noise of each algorithm can affect the overall performance of the network. The chosen algorithms are adequate for providing significant insight into the interoperability among different algorithms.

A register Minimum Explicit Rate (MER) and two flags MCI and MNI are widely used in proposed algorithms. MER stores the minimum of the ER's values among those indicated by the BRM cells received from the branches and the MACR (Mean Allowed Cell Rate) calculated locally at the branch-point on the reception of each FRM cell. MCI and MNI values are updated by doing 'OR' operation with their respective values carried by a currently received control BRM cell and locally calculated CI and NI values for the branch-point. MER is initialized to peak cell rate whereas MCI and MNI are initialized to zero.

3.4.1 Wait-for-all Algorithm

The wait-for-all consolidation algorithm [60] called in this thesis as Ren. A branch point waits for BRM cell from all branches. The branch point returns a BRM cell with the completed congestion information to its upstream node (or source) whenever it receives a first FRM cell after having completely consolidate the BRM cells. In addition to MER, MCI and MNI, the algorithm requires two more counters, which are Number_of_BRMreceived and Number_of_Branches. The first one is used for counting the number of branches from which BRM cells have been received at a branch point (after the last BRM was sent by the branch point) and the second one is used for storing the number of branch connecting to this branch point. Moreover, a BRMReceived flag is needed for each branch to indicate whether a BRM cell has been received from this particular branch, after the last BRM cell was passed.

In multicast point-to-multipoint ABR, it is important to consolidate the congestion feedback at each branch point and only one consolidated feedback is sent upstream to avoid feedback implosion. Since there are many downstream branches, RM cell may arrive at the branch point at significantly different time. Hence, the consolidation of feedback RM cell must be synchronized at the branch point before the consolidated RM cell can be sent upstream. We called this operation as RM cell synchronization. Ren algorithm perfectly poses this characteristic. However, this algorithm is not achievable for the response time minimization characteristic, which is the ability to minimize the response time of the consolidation algorithm. Although this algorithm reduces consolidation noise, it exhibits a considerably slow transient response.

The pseudo code and flowchart of the algorithm are shown in Figure 3.2 and Figure 3.3, respectively.

On the receiving of a FRM cell:

1. Multicast this FRM cell to all branches;

On the receiving of a BRM cell from branch i:

1. IF NOT BRMReceived(i) THEN
    A. Let BRMReceived(i) = 1;
    B. Let NumberOfBRMsReceived = NumberOfBRMsReceived + 1;
2. Let MER = Min (MER, ER from BRM cell), MCI = MCI OR CI from BRM cell and MNI = MNI OR NI from BRM cell;
3. IF NumberOfBRMsReceived is equal to NumberOfBranches THEN
    A. Let MER = Min (MER, minimum ER calculated by ERICA for all branches);
    B. Return the BRM cell with ER = MER, CI = MCI and NI = MNI to the root;
        C. Let MER = PCR, MCI = 0 and MNI = 0;
        D. Let NumberOfBRMsReceived = 0;
        E. Let BRMReceived(i) = 0 for all branches;

    ELSE

        A. Discard this BRM cell;

Figure 3.2 Pseudo code of wait-for-all algorithm

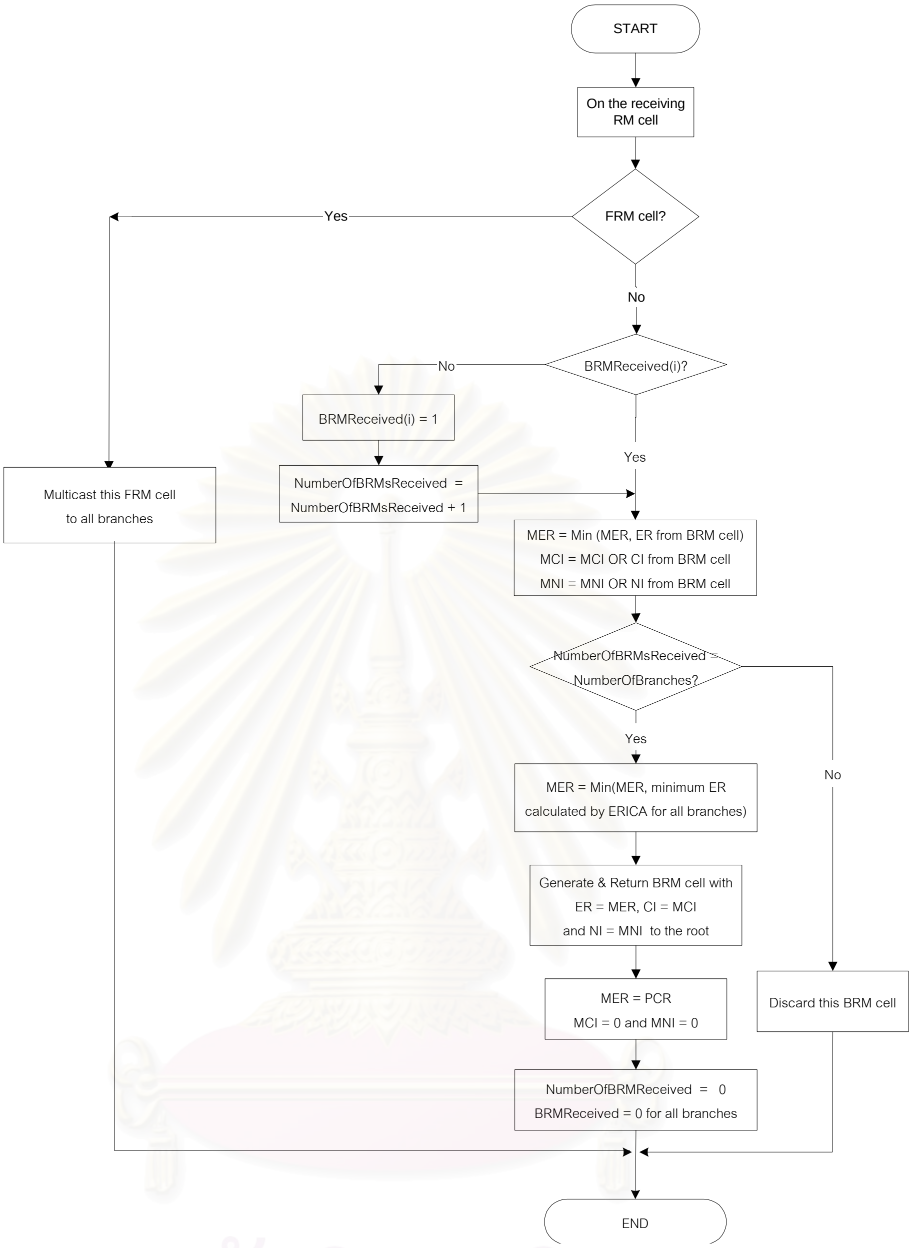


Figure 3.3 Flowchart of wait-for-all algorithm

3.4.2 Not wait-for-all Algorithm

The not wait-for-all consolidation algorithm [22] called in this thesis as R-S. The basic idea of these algorithms, as shown in Figure 3.4 and Figure 3.5, on the reception of FRM cell when at least one BRM has been received from its branch, the branch point generates BRM cell and immediately returns a BRM cell to the source. In addition to MER, MCI and MNI, there is one more flag for AtLeastOneBRM which is stored for each point-to-multipoint VC. This flag is set to 1 on reception of a BRM cell and is reset to zero vice versa. . This is to minimize the transient response time of the algorithm and makes the branch point feedbacks the congestion information faster than that of Ren, however, because the lacking of a RM cell synchronization the consolidation noise is considerable high.

On the receiving of a FRM cell:

1. Multicast this FRM cell to all branches;
2. IF AtLeastOneBRM THEN
   A. Let MXER = ER from FRM cell, MXCI = CI from FRM cell
      and MXNI = NI from FRM cell;
   B. Let MER = Min (MER, minimum ER calculated by ERICA for all branches);
   C. Return the BRM cell with ER = MER, CI = MCI and NI = MNI to the root;
   D. Let MER = MXER, MCI = MXCI and MNI = MXNI;

On the receiving of a BRM cell from branch i:

Let MER = Min (MER, ER from BRM cell), MCI = MCI OR CI from BRM cell and MNI = MNI OR NI from BRM cell;

AtLeastOneBRM = 1;

Discard this BRM cell;

Figure 3.4 Pseudo code of not wait-for-all algorithm

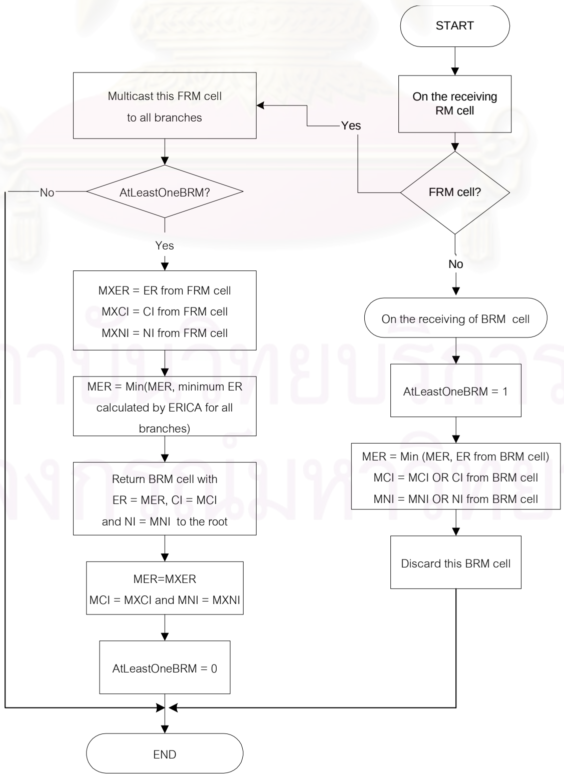


Figure 3.5 Flowchart of not wait-for-all algorithm

3.4.3 Fahmy Algorithm

There are some proposed algorithms to achieve a fast response with low consolidation noise [4].The branch point has to wait for all BRM cells from all branches like the wait-for-all. If there is an overloaded condition say, the rate in current BRM cell is much less than last BRM cell, exists at the branch point. The algorithm can detect this situation and a branch point immediately feedbacks the BRM cell with the low ER value to the source to avoid a data overflow. The branch point will send the BRM back again when BRM cell from all branches have been arrived. With this approach, we notice that the branch point seems to be inactive during waiting for all BRM cells. Hence, this will slow down the branch point's response. Moreover, during this period if the network has more bandwidth available due to the off period of Variable Bit Rate (VBR) traffic. This amount of bandwidth can not be used and will result in a lower network utilization.

There are some proposed algorithms to achieve a fast response with low consolidation noise [57]. The authors in [57] define the overload and underload conditions of a branch. The branch point has to wait for all BRM cells from all branches like the wait-for-all. This means that Fahmy has a RM cell synchronization (during the network is in a normal or underload situation). If there is an overloaded condition for example, the rate in current BRM cell is much less than last BRM cell, exists at the branch point. The algorithm can cope with this event by invokes the immediate rate calculation option (Using ERICA algorithm [51] , which is an algorithm used in simulations to calculate the fair share explicit rate feedback based on the load at each port) and then immediately feedbacks the BRM cell with the low ER value to the source to avoid data overflow causing from waiting feed back from all branches. This means the Fahmy is now working in the response minimization mode. The branch point will send the BRM back again when BRM cell from all branches have been arrived. Hence, obviously that this algorithm has both RM cell synchronization and response time minimization characteristics. However, with this approach it introduces a relatively high complexity and we notice that the branch point seems to be inactive during waiting for all BRM cells. Hence, this will slow down the branch point's response. Moreover, during this period if the network has more bandwidth available due to the off period of VBR traffic. This amount of bandwidth can not be used and will result in a lower network utilization. The details of this algorithm are demonstrated in Figure 3.6 and Figure 3.7.

On the receiving of a FRM cell:

Multicast this FRM cell to all branches;

Let FRMMinusBRM = FRMMinusBRM + 1;

On the receiving of a BRM cell from branch i:

1. Let SendBRM = 0;
2. Let Reset = 1;
3. IF NOT BRMReceived(i) THEN
   A. Let BRMReceived(i) = 1;
   B. Let NumberOfBRMsReceived = NumberOfBRMsReceived + 1;
4. Let MER = Min (MER, ER from BRM cell), MCI = MCI OR CI from BRM cell and MNI = MNI OR NI from BRM cell;
5. Let MER = Min (MER, minimum ER calculated by ERICA for all branches);
6. IF MER $\geq$ LastER AND SkipIncrease > 0 AND NumberOfBRMsReceived is equal to NumberOf Branches THEN
   A. Let SkipIncrease = SkipIncrease – 1;
   B. Let NumberOfBRMsReceived = 0;
   C. Let BRMReceived(i) = 0 for all branches;

   ELSE IF MER < LastER * Threshold THEN
   A. IF NumberOfBRMsReceived < NumberOfBranches THEN
      1. Let SkipIncrease = SkipIncrease + 1;
      2. Let Reset = 0;
   B. Let SendBRM = 1;

   ELSE IF NumberOfBRMsReceived is equal to NumberOfBranches THEN
   A. Let SendBRM = 1;
7. IF SendBRM THEN
   A. Let MER = Min (MER, minimum ER calculated by ERICA for all branches);
   B. Return the BRM cell with ER = MER, CI = MCI and NI = MNI to the root;
   C. Let LastER = MER;
   D. IF Reset THEN
      1. Let MER = PCR, MCI = 0 and MNI = 0;
      2. Let NumberOfBRMsReceived = 0;
      3. Let BRMReceived(i) = 0 for all branches;
      E. Let FRMMinusBRM = FRMMinusBRM – 1;

   ELSE
      A. Discard this BRM cell;

Figure 3.6 Pseudo code of Fahmy algorithm

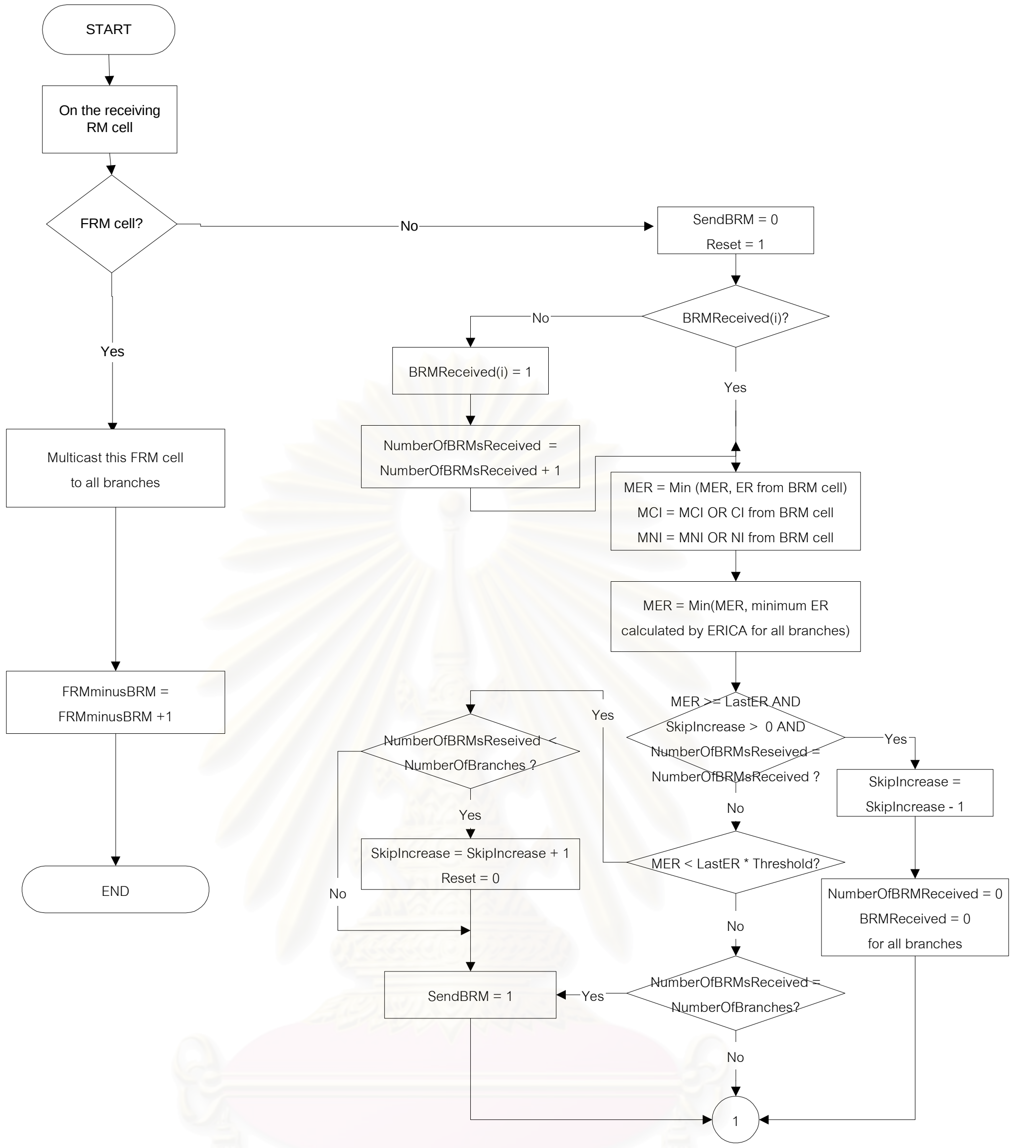


Figure 3.7 Flowchart of fahmy algorithm

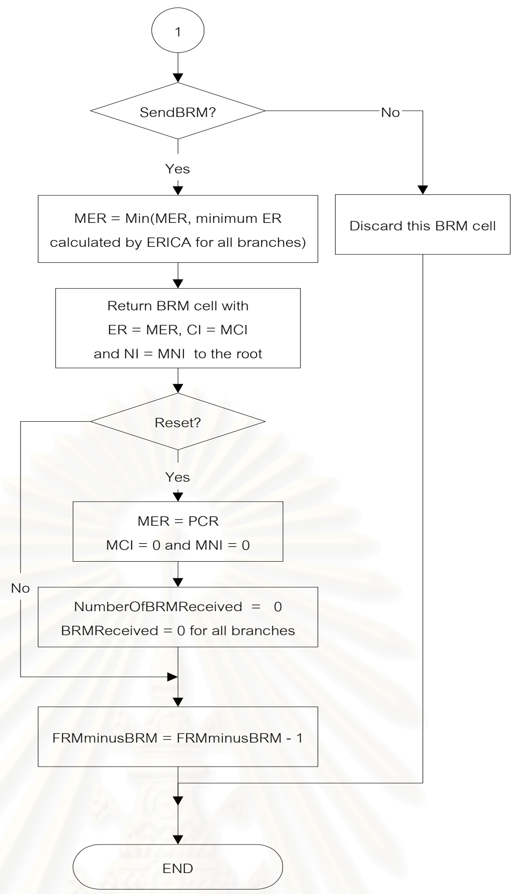


Figure 3.7(Continue) Flowchart of Fahmy algorithm

CHAPTER 4

Proposed Consolidation Algorithms

4.1 RQB

In this chapter, a new consolidation algorithm called Rate-Queue Balanced (RQB) algorithm [42] is presented. The RQB was designed to compromise between the response time and accuracy. It does not only take care of overload condition in downstream branches but also has the potential to detect the overload situation at the branch point itself [57]. It can track, during the transient state, the changing rate in the network due to i.e. VBR source without consolidation noise. It is also capable to adapt itself to work in fast response or high accuracy mode. The algorithm works as follows.

Upon the receipt of the BRM cell at the branch point, the branch's status flag is set. This means that this branch has already received the BRM. At each branch of the branch point, there are some parameters, MER and MQL registers and MCI and MNI flags. Initially, MER is equal to PCR, MQL is equal to zero, and the MCI and MNI flags are reset. The network parameters (ER,QL,CI and NI) from the BRM are stored in MER and MQL registers and MCI and MNI flags, respectively, instead of the previous value. The ER value that will be sent to the source is analyzed by comparing for the smallest value among the MER register and ER_calculated by ERICA algorithm for all branches and stored in THIS_ER register. With the same manner, for QL, the algorithm stores the biggest value among MQL and queue length for all branches at that time in THIS_QL register. For CI and NI, an 'OR' operation is employed. At this point we have all important parameters to be sent back as RM cell (RM(ER,QL,CI,NI)) to control the source. The algorithm checks from the branch's status flag whether the THIS_ER and THIS_QL it has now is gathered from all branches or not. If it is the case, it means that the branch point has got all information from all branches so that it will inform the source to generate a traffic that is not overloaded to any branch. In this situation there will be no consolidation noise occurred. In the case that the branch point does not gather BRM cell from all branches, the algorithm grows through condition b.1-b.3 in Figure 4.1.

Rate-Queue Balanced Pseudo Code

```
Upon the receipt of forward RM(ER, CI, NI, QL) cell :
  Multicast this RM cell to all participating branches
Upon the receipt of backward RM(ER, CI, NI, QL) cell from branch j:
  1. Let Branch's status flag(j) = Branch's status flag(j) + 1
  2. Let MER(j) = ER_FROM_BRM, MQL(j) = QL_FROM_BRM,
     MCI(j)  = CI_FROM_BRM, and MNI(j) = NI_FROM_BRM
  3. Let THIS_ER = min(minimum MER for all branches, minimum ER
     calculated by rate allocation algorithm for all branches),
     THIS_QL = max(maximum MQL for all branches, Queue at the
                         branch point),
     CI = OR (MCI for all branches),  and  NI = OR (MNI for all branches)
  4a IF (Branch's status flag  > 0 for all branches) THEN
        Send  RM(ER, CI, NI, QL) cell back to source
        Let LAST_ER = THIS_ER, and LAST_QL = THIS_QL
Reset BRANCH'S STATUS FLAG = 0 for all branches
  4b  ELSE
        b.1     IF ((LAST_ER > THIS_ER) AND (LAST_QL < THIS_QL))
                THEN
        Let  RAI = High Value
        b.2     ELSE IF ((LAST_ER < THIS_ER) AND (LAST_QL > QL))
                THEN
                 Let  RAI = Low Value
        b.3     ELSE
        Let  RAI = Medium Value
    b.4     IF (RandomValue < RAI) THEN
       Send  RM(ER, CI, NI, QL) cell back to source
                 Let  LAST_ER = THIS_ER, and  LAST_QL = THIS_QL
       b.5     ELSE
                   Discard  RM cell
```

Figure 4.1   Pseudo code of Rate-Queue Balanced algorithm

In condition b.1, if THIS_ER has a smaller value than the previously sent ER (LAST_ER) and THIS_QL has a bigger value than the previously sent QL (LAST_QL). It is assumed that the downstream network may have some congested point then the algorithm set the Response-Accuracy Index (RAI) at a high value nearly 1 (the value of RAI range from 0 to 1). This means we now work in a fast response mode. Condition b.2 is a complementary case of b.1. This indicates that the network does not have any congestion elsewhere in downstream nodes and has a high chance to be discarded and the branch point has to collect more network information. Thus RAI is set to a low value nearly 0. It is then assumed that we are now working in a high accuracy mode. In condition b.3, RAI is set to a medium value because there is only one register (not both) of THIS_ER and THIS_QL indicated the bottleneck link elsewhere in the downstream network. So this RM has an equal chance to be discard or to be sent back to the source. This mode is called a medium response mode. As Fahmy, RQB poses both RM cell synchronization and response time minimization characteristic. The flowchart of RQB algorithm is shown in Figure 4.2.

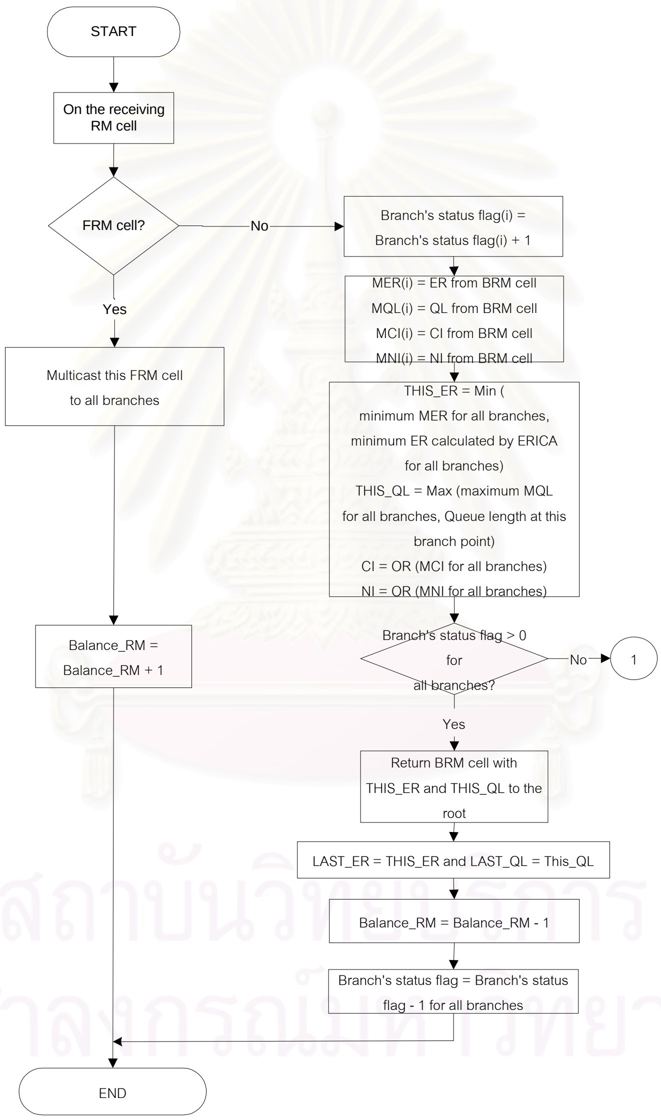


Figure 4.2 Flowchart of RQB algorithm

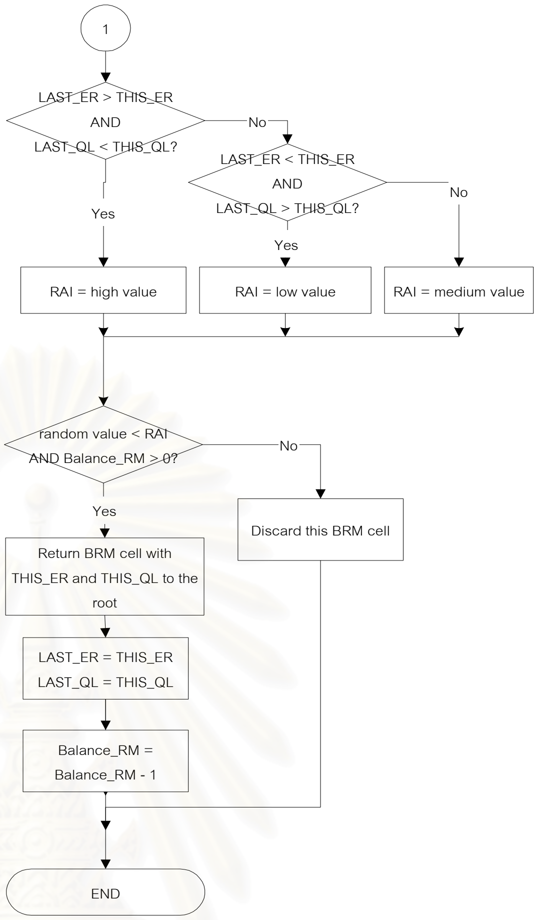


Figure 4.2(Continue) Flowchart of RQB algorithm

4.2 Selective BRM Feedback (SBF) Algorithm

In this section, an efficient consolidation algorithm is presented to overcome the drawbacks of the previously proposed papers. We have learnt that if the branch point does not wait for all BRM cells it will introduce the noise. On the other hand, the response is very slow if it waits for all. Based on the fact that the branch point has to send the least ER value among all BRM cells from its branches to the source. Hence, we design a consolidation algorithm called Selective BRM Feedback (SBF) [43] that feedbacks the BRM cell to the source selectively. The branch point does not only wait for all BRM cells but also select the BRM cell which contains the least ER value to send to the source. This means that SBF has a response time minimization characteristic. For RM cell synchronization, SBF virtually poses this characteristic since it does not physically wait for RM cell from all branches to collect the most congested information but it logically do this job by monitoring and tracking the branch that most congested. The difference between SBF and not wait-for-all is that during an inter-FRM interval SBF may feedback a single or multiple BRM to the source depending on the traffic condition in the network at that time. This means that the source, if necessary, can get the information from the network more frequently. The SBF is designed to achieve a fast response, low consolidation noise and low complexity. It uses 3 registers at the branch point. An MER register is used for storing the ER value, a Branch_number register is used for storing the number of branch from which the branch point receives the BRM cell and a Balance_RM is used for controlling the BRM to FRM ratio to be 1. The algorithm can take care of overload condition in downstream branches and can utilize, during the transient state, the available bandwidth, especially left from the VBR source in the network without consolidation noise. It is also insensitive to the number of branches and branch point levels in the network. The algorithm works as in Figure 4.3.

```
Proposed consolidation algorithm pseudo code
Upon the receipt of forward RM(ER, CI, NI) cell :
  1. Multicast this RM cell to all participating branches
  2. Let Balance_RM = Balance_RM + 1
Upon the receipt of backward RM(ER, CI, NI) cell :
    IF (Branch_Number == i) THEN
        IF   (ER from BRM == MER && Balance_RM ≤ 0) THEN
             Discard RM cell;
        ELSE    Update MER, Branch_Number and send back to the source by let
                { MER = ER from BRM;
                  Branch number = i;
                  ER = min(ER from BRM, minimum ER calculated by rate allocation algorithm for all branches);
                  CI = CI from BRM, and NI = NI from BRM;
                  Send RM(ER, CI, NI) cell back to source;
                  Let Balance RM = Balance_RM – 1;
             }
    ELSE
    IF   (ER from BRM < MER && Balance_RM ≤ 0) THEN
             { Update MER, Branch_Number and send back to the source }
         ELSE Discard RM cell;
```

Figure 4.3   Pseudo code of the proposed algorithm

Upon the receipt of the BRM cell, the branch point checks the branch number and the ER value. If the branch number is different from the current branch number stored in Branch_Number register and the received ER value is less than the current ER value stored in MER register, the

MER and Branch_Number are updated to the new values. Otherwise, the received BRM cell is discarded. In case BRM cell comes from the same branch number stored in the Branch_Number register, the branch point always updates the ER value in MER, no matter it is more or less with respect to the previous one. Thus, these registers always preserve the latest congestion information in the downstream branches. From the above explanation, alternatively, it can be concluded into four cases.

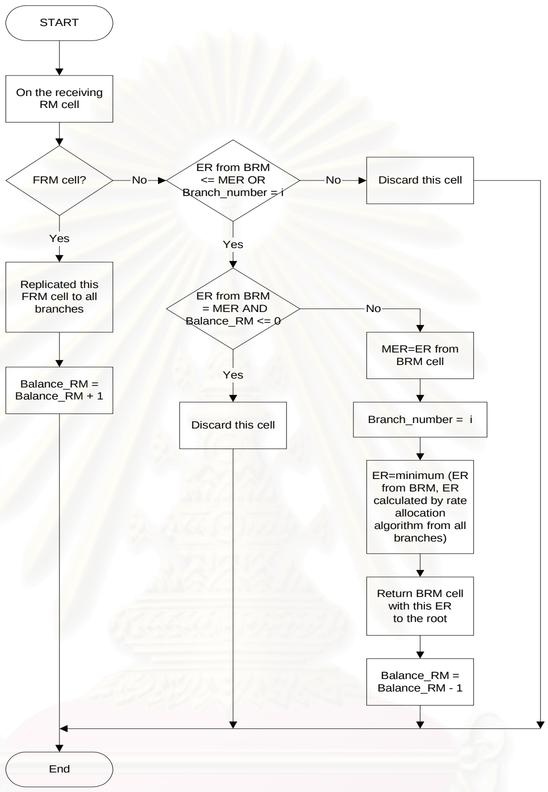


Figure 4.4 Flowchart of the proposed algorithm

Case 1: if (ER from BRM < MER and Branch_Number = i) → Update MER and Branch_Number

Case 2: if (ER from BRM > MER and Branch_Number = i) →Update MER and Branch_Number

Case 3: if (ER from BRM < MER and Branch_Number ≠ i) →Update MER and Branch_Number

Case 4: if (ER from BRM > MER and Branch_Number ≠ i) → Not update MER and Branch_Number

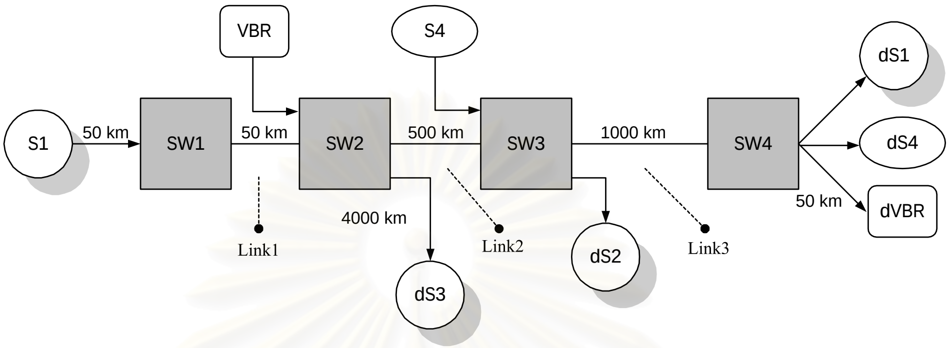
We can see that in case 1 and case 3 (ER from BRM < MER) the MER is always updated. This is designed to feedback the lower value of ER from BRM quickly to the source. This technique statistically reduces the waiting time of the branch point i.e. if the branch that has the lowest ER is located nearest to the branch point, the branch point does not to wait for consolidating BRM from all branches. In case 2, where the value of ER from BRM is larger than MER and the Branch_Number is equal to i. The branch point also sends the updated ER to source. This is our intention to let the branch point to have a feature that can utilize the bandwidth left in this lowest branch. For example, if this branch shares the bandwidth with the VBR traffic and in this epoch the VBR traffic is active, the traffic available for ABR service in this branch is the traffic that left from VBR. For the next epoch (normally equal to the Average Interval (AI) time in ERICA), if the VBR traffic is off. The available bandwidth for ABR is increased and branch point can recognize this changing bandwidth and tell the source to increase its rate. For case 4, the branch point discards the BRM cell because branch number i is the branch that can support the lowest rate (has the least bandwidth). The branch point will be suffered from the buffer overflow if we update the MER with the ER from BRM from the branches that have a higher rate. With this approach, the branch point is updated by the BRM cell from the most congested branch and sends it back to the source promptly without waiting for all BRM cells from all branches (that some of them might be non-responsive). The salient features of SBF are a fast response of branch point while a low consolidation noise condition is preserved. In addition, it has a low implementation complexity (only 3 registers are used while the previously proposed algorithms used more registers and some additional flags). The flowchart of the SBF consolidation algorithm is shown in Figure 4.4.

4.3 Simulation Results

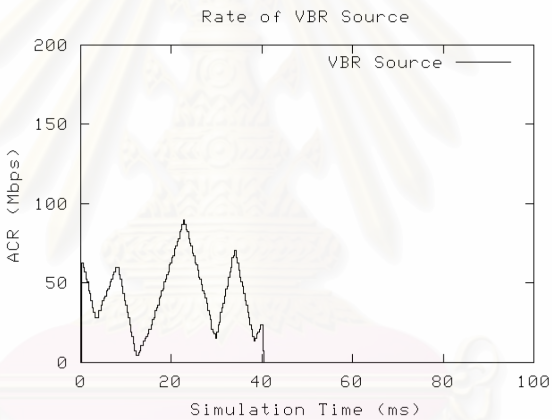
Simulation of the aforementioned algorithms is given in this section. The followings are examples of application of the proposed algorithms compared to other well-known algorithms in various environments. There are many algorithms to be investigated including our two proposed algorithms. For the purpose that all the previously proposed algorithms and our proposed ones to be clearly compared, only SBF algorithm is selected in simulation. RQB is omitted for the simulation because it shows almost the same result as SBF does, but with more in complexity.

4.3.1 Bursty VBR Configuration

For convenience, we will refer to the wait-for-all, the not wait-for-all, and Fahmy algorithm as Ren, R-S and Fahmy, respectively and our proposed algorithm as SBF.



(a) Network configuration



(b) VBR traffic

Figure 4.5 Bursty VBR network configuration

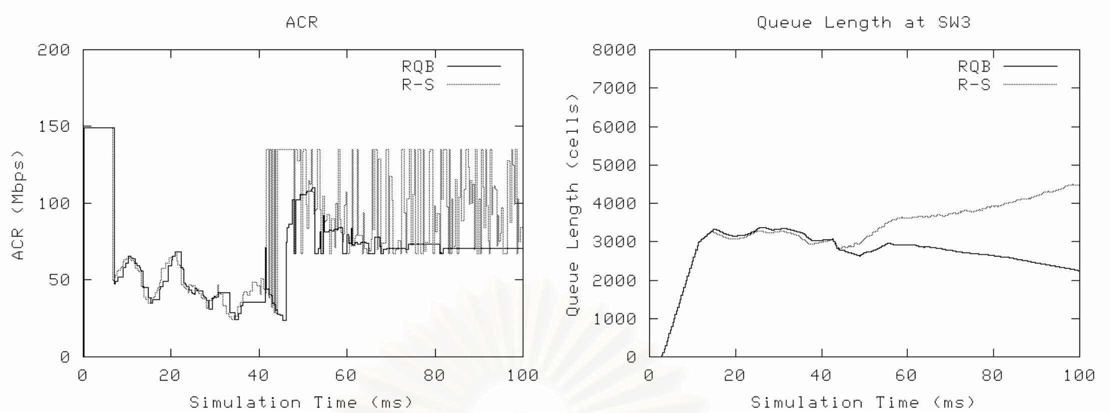
Parameters Setting

We use these values in every Network Model except where specified

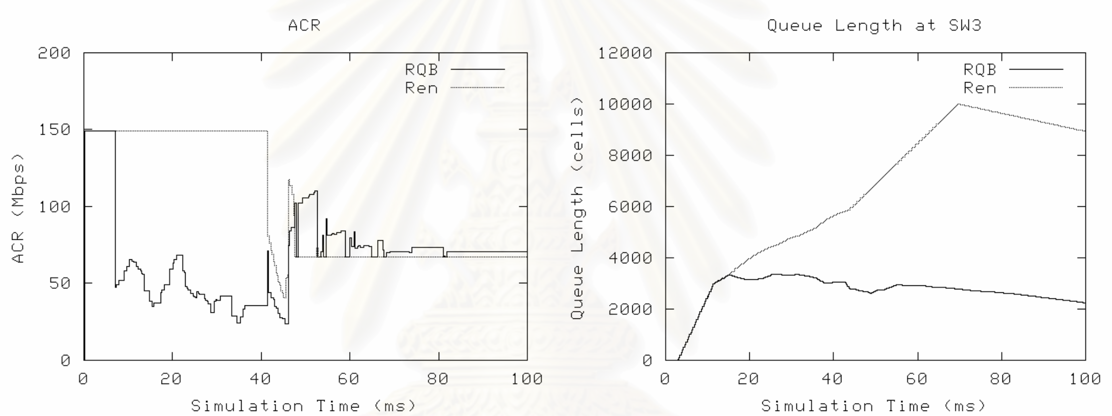
- Except where indicated, all link capacities are 150 Mbps.
- All switch-to-end system links are 50 km except where specified.
- The switch target utilization is set to 90%.
- The switch averaging interval is set to 0.3 ms.
- The parameter Transient Buffer Exposure (TBE) is set to large values for preventing the rate decrease.
- All sources are persistent sources.

- The source parameter Rate Increase Factor (RIF) is set to 1.
- The Peak Cell Rate (PCR) is set to 150 Mbps. The Initial Cell Rate (ICR) is set to 150 Mbps and 5 Mbps for High ICR and Low ICR, respectively.
- Use ERICA switch algorithm with Max-Min fairness as rate allocation algorithm.

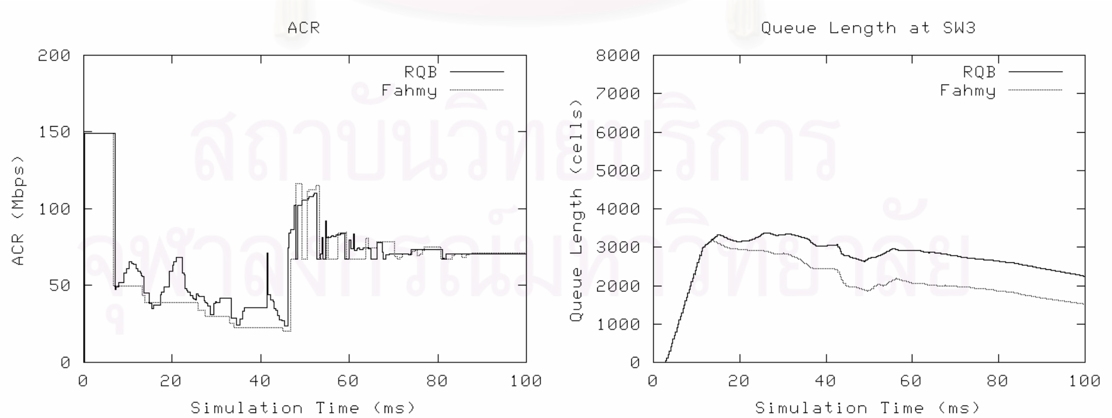
In this section, we will show the simulation results of the SBF comparing to the previously proposed algorithms. The network configuration is shown in Figure 4.5a. Source S1 and its destinations: dS1, dS2 and dS3 are formed a multicast session. The maximum round trip delay ($t_{RTDmax.}$) in a multicast session is 41 ms on S1 and dS3 path. We also put a unicast session to run as background traffic in the network configuration. S4 and VBR source is a point-to-point connection a nd has a dS4 and dVBR destination, respectively. The VBR traffic source is shown in Figure 4.5b. The simulation time shown in every figure throughout this thesis means a network virtual time. It is not a real time. That is 100 ms simulation time may take few minutes real time. For simplicity in explaining the results, we would consider the response of the network separately into two states. The first state lies between 0-41 ms is a transient state and the second state: the steady state is from 41 ms onwards. Figure 4.6a and Figure 4.6d shows the response of R-S and SBF algorithm. The outstanding characteristic of R-S and SBF algorithm is the fast transient response. During transient state, we see that at time 6.5 ms, the allowed cell rate of the source or ACR of S1 drops from 150 Mbps to a fair share value of available bandwidth left in Link2. It is because the first BRM cell arrives at S1 is from dS2, which located 650 km apart (we use 5 microseconds per kilometer delay so the RTD for dS2 is about 6.5 ms). Due to the VBR background traffic, the available bandwidth in the network is changed all the time. However, the simulation result shows that both R-S and our proposed RQB and SBF can track this changing traffic and can control S1 to deliver the cell rate correspondingly to available resources. During steady state, we see clearly that for RQB and SBF, the ACR of S1 converges to a fair share value with no noise while for R-S, a consolidation noise is occurred. In Figure 4.6b and Figure 4.6e, contrarily to R-S algorithm, Ren has a slow transient response but exhibits no consolidation noise. S1 will drop its ACR to the fair rate when BRM from the farthest destination has arrived. While RQB and SBF response time is only 6.5 ms, Ren takes 41 ms to response. During this period, S1 delivers the cell at rate 150 Mbps to its downstream nodes. Hence, the queue is explosively built up at cell-traversed switches. For consolidation noise, Ren, RQB and SBF exhibit a good characteristic that no noise is introduced. In Figure 4.6c and Figure 4.6f, we can see that Fahmy has a fast response and low consolidation noise as RQB and SBF. But during the transient state, Fahmy can detect the overload in SW3 at the time 6.5 ms. It suppresses the ACR of S1 to a low value and is not able to track and utilize the available bandwidth. This leads to a low utilization of Link1. Obviously, RQB and SBF outperform the others in terms of response time and consolidation noise. However, the average queue length of the SBF is a bit higher than that of Fahmy.



(a) R-S and RQB

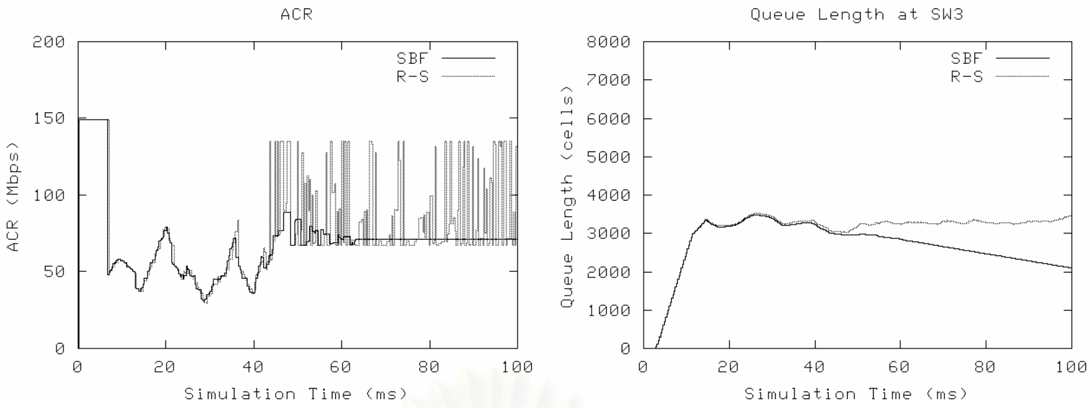


(b) Ren and RQB

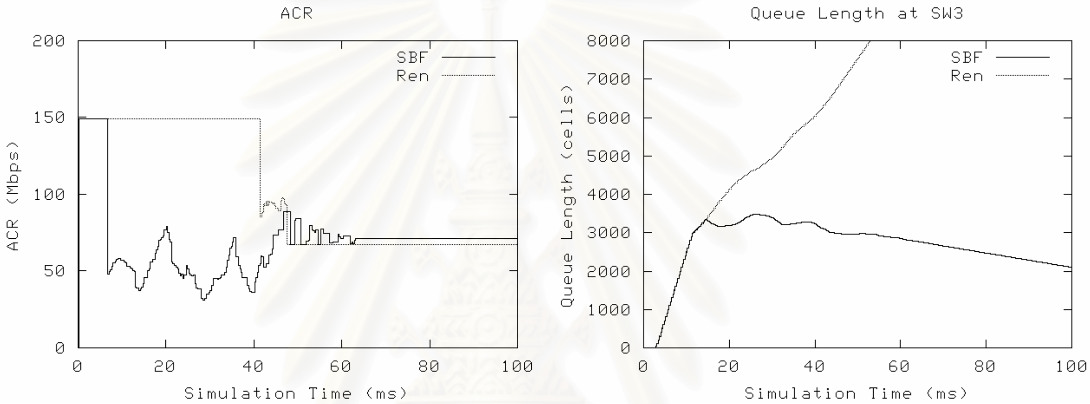


(c) Fahmy and RQB

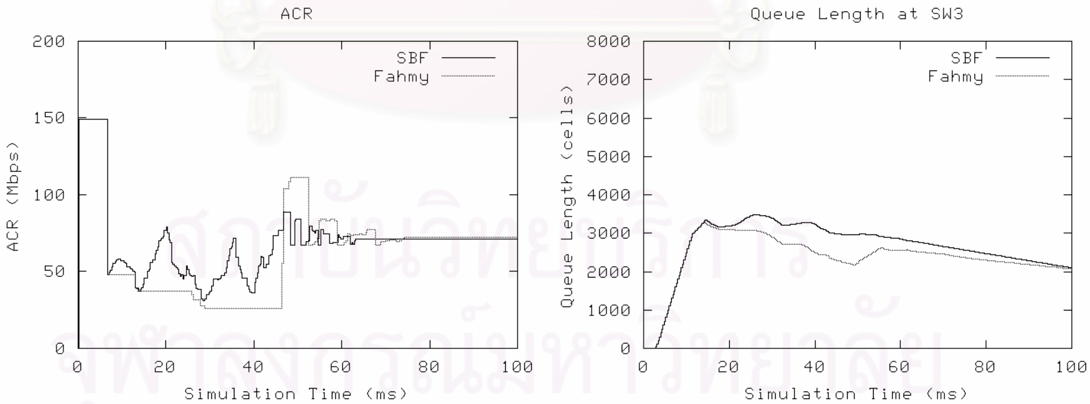
Figure 4.6 Performance comparison of RQB, SBF and others.



(d) R-S and SBF



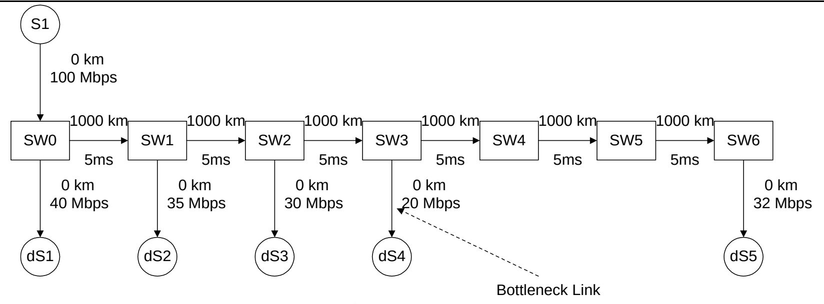
(e) Ren and SBF



(f) Fahmy and SBF

Figure 4.6(Continue) Performance comparison of RQB, SBF and others.

4.3.2 GT Chain Configuration



GT Chain Configuration [59]

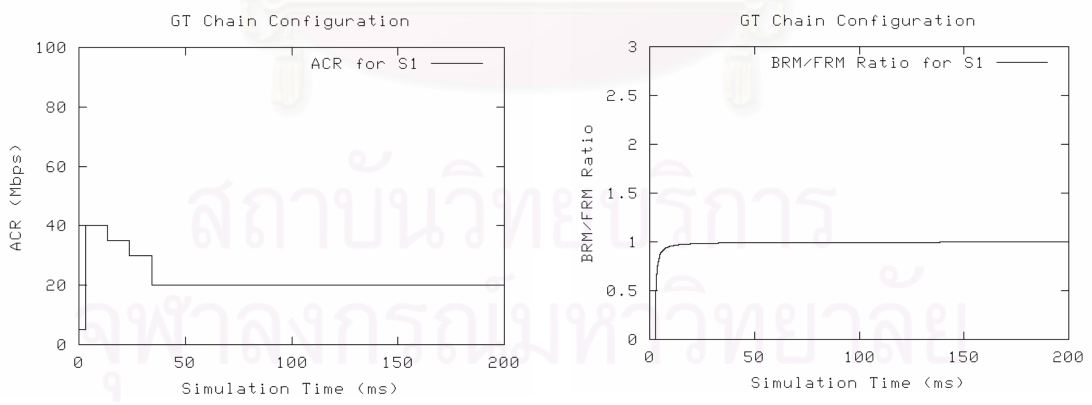
(Transient Response and BRM/FRM Cell Ratio Test)

All switch-to-end system links are 0 km, All link capacities are 100 Mbps, RIF = 1

Utilization Factor = 1.0 (100%), PCR = 100 Mbps, ICR = 5 Mbps, AI = 100 cell-time

Figure 4.7 GT Chain network configuration

This network model is modified from Chain configuration proposed by Jiang [59]. The link between the switches is 1,000 km long. We assume that the end systems are co-located with the switch (no distance apart). The round trip delay of the cell from the source to dS1, dS2, dS3, dS4 and dS5 are 0 ms, 10 ms, 20 ms, 30 ms and 60 ms, respectively. S1 sends data at a rate of 5 Mbps for ICR and 100 Mbps for PCR. The purpose of simulating this network model is to evaluate the performance of the consolidation algorithms when using the low ICR source. It is also used for investigating the BRM/FRM ratio for each algorithm.

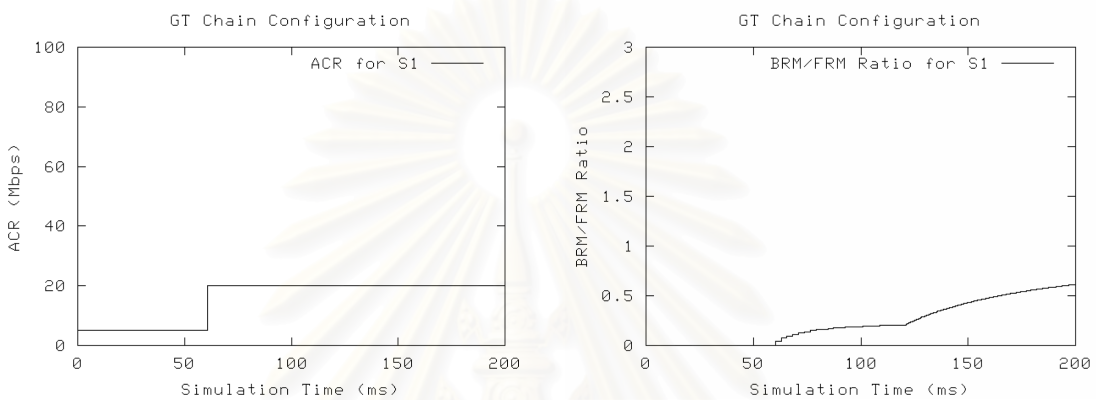


(a) ACR

(b) BRM/FRM Ratio

Figure 4.8 Simulation results of GT chain configuration with R-S algorithm

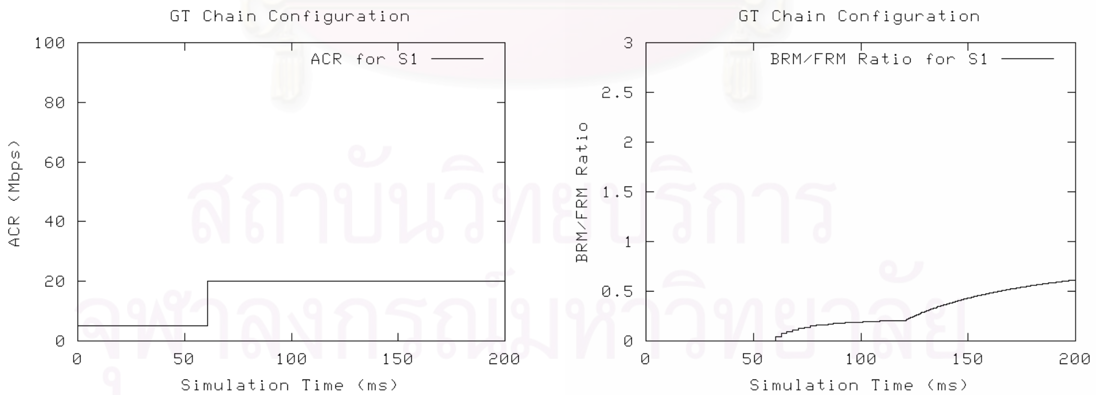
In Figure 4.8a, we see that S1 response to the network by adjusting the ACR very quickly. Because the R-S algorithm will send the BRM back to the source when at least one BRM cell is received at the branch point. Hence, when SW0 get the first FRM from the source there is no BRM available. SW0 has to wait for another BRM from the network which is dS1: the nearest end system. SW0 is able to feedback BRM to the source whenever the next FRM has arrived at and this will take 32 cell time (the number of resource management cell spacing or Nrm). We set the ICR of the source to 5 Mbps then the source takes 2.7 ms ($32*424/(5*10^6)$) time delay before adjusting its rate according to the network information. As described above, the BRM/FRM ratio at the beginning is less than one and approaching one eventually as shown in Figure 4.8b



(a) ACR

(b) BRM/FRM Ratio

Figure 4.9 Simulation results of GT chain configuration with Ren algorithm

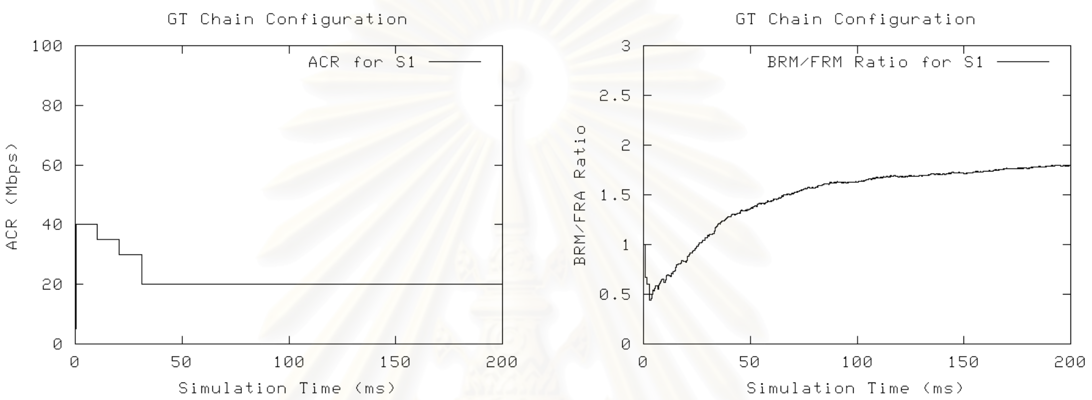


(a) ACR

(b) BRM/FRM Ratio

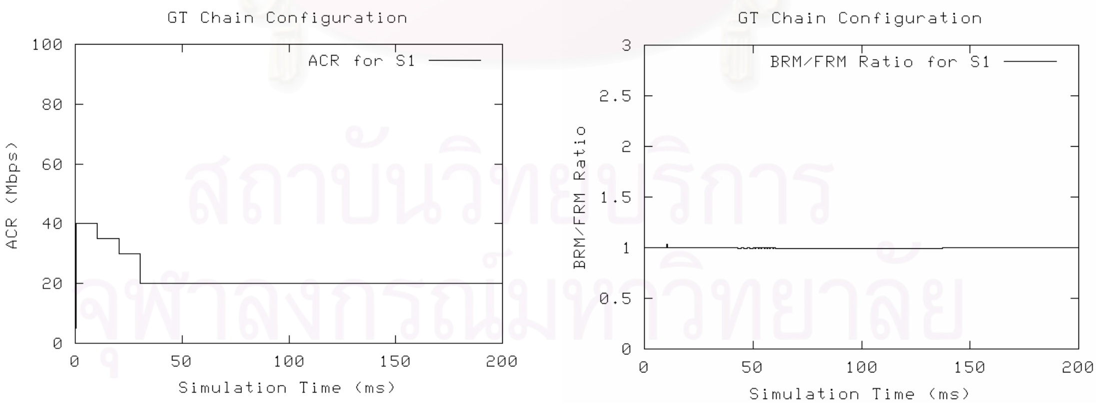
Figure 4.10 Simulation results of GT chain configuration with Fahmy algorithm

For the Ren algorithm, S1 cannot increase its ACR to the value suggested by the network until it received the BRM cell from all destinations. As shown in Figure 4.9a, the source takes 60 ms for the idle state (no feedback information feedback to the source because SW0 has to collect all BRM). This results a slow response time of the source to the network. The BRM/FRM ratio is zero (no BRM cell) during the transient state as shown in Figure 4.9b. For Fahmy algorithm, it results exactly the same result with Ren as shown in Figure 4.10a and Figure 4.10b. This is because Fahmy adopt the Ren algorithm as its main operation but adds up with the branch point overload detection feature. There is no overload at SW0 for this network configuration. Hence, there is no difference in this two algorithms. We will see the advantage of Fahmy over the Ren in the case that there is a congested condition at the branch point.



(a) ACR (b) BRM/FRM Ratio

Figure 4.11 Simulation results of GT chain configuration with RQB algorithm

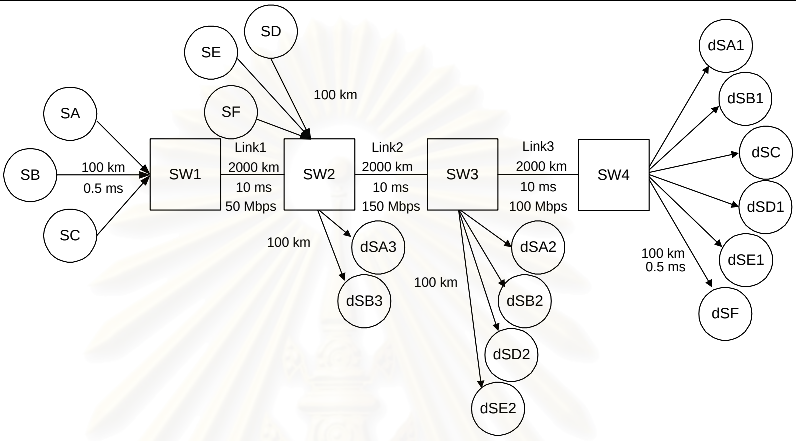


(a) ACR (b) BRM/FRM Ratio

Figure 4.12 Simulation results of GT chain configuration with SBF algorithm

Figure 4.11 and Figure 4.12 show the result of RQB and SBF algorithm, respectively. For an ACR of the source, both algorithm exhibit the similar result. However, SBF has been designed to control the BRM/FRM ratio. Hence, this ratio is always controlled to be one as shown in Figure 4.12b. SBF exhibits a very fast response time because the algorithm continuously feedbacks the BRM cells from the most congested branch or the branch which has the lowest ER value, it is not necessary to waste the time collecting BRM from all branches or waiting for the arriving of the FRM cell before sending the BRM back.

4.3.3 Source Bottleneck Configuration



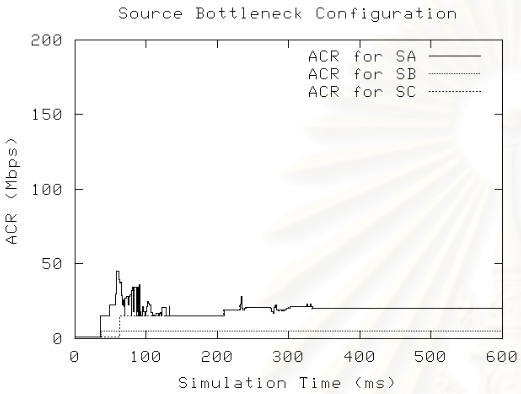
<u>Source Bottleneck Configuration</u>
(Fairness and Transient Response Test)
SB & SE: Source Bottleneck at Rate = 5 Mbps
SA, SB & SC: ICR = 1 Mbps
SD, SE & SF: ICR = 150 Mbps
All switch-to-end system links are 100 km, All link capacities are 150 Mbps
Target Utilization = 0.90 (90%), RIF = 1, AI = 1 ms

Figure 4.13 Source bottleneck network configuration

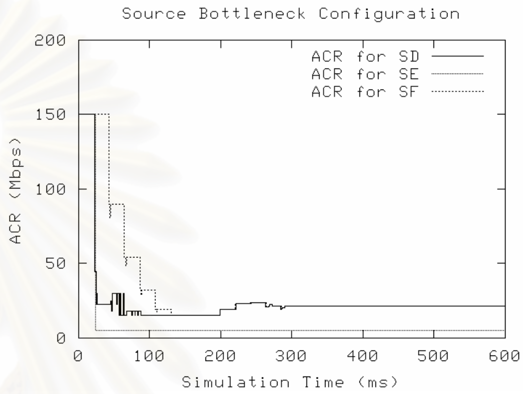
In this configuration we intend to test the performance of different consolidation algorithms in the situation that there is a source bottleneck in the network. The source bottleneck means that the source sends the data at the rate less than the fair share calculated by the Max-Min fairness scheme. The good consolidation algorithm should be able to fairly utilize this rest bandwidth efficiently by re-calculating and assign to the non-bottleneck sources or other network elements.

The sources SA, SB, SD and SE and their destinations are the multicast sessions while SC and SF are the unicast sessions. If there is no source bottleneck at source SB and SE, source SA, SB and SC must occupy a bandwidth of Link1 equally at (50*0.9)/3 = 15 Mbps each (the Link 1 bandwidth is 50 Mbps, the utiliation is 90 %). While a fair share bandwidth of SD, SE and SF through Link 3 are equal to ((100*0.9)-45)/3 = 15 Mbps each.
Because the data rate sending by SB and SE is limited to 5 Mbps which is lower than the fair share rate. The rest bandwidth should be shared to the others. Then the new fair share bandwidth of each source should be:

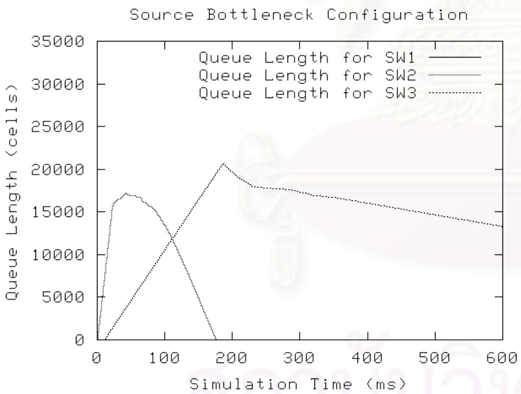
- SA should get ((50*0.9)-5)/2 = 20 Mbps.
- SB is limited to 5 Mbps.
- SC should get ((50*0.9)-5)/2 = 20 Mbps.
- SD should get ((100*0.9)-45-5)/2 = 20 Mbps.
- SE is limited to 5 Mbps.
- SF should get ((100*0.9)-45-5)/2 = 20 Mbps.



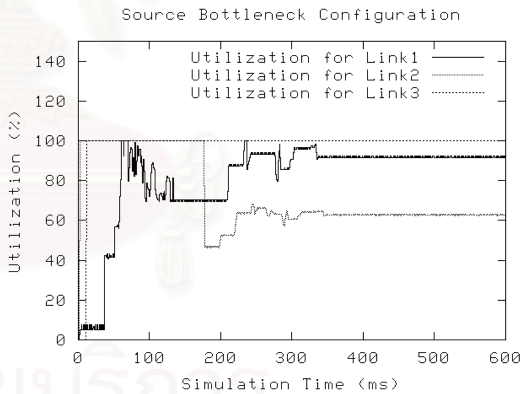
(a) ACR



(b) ACR

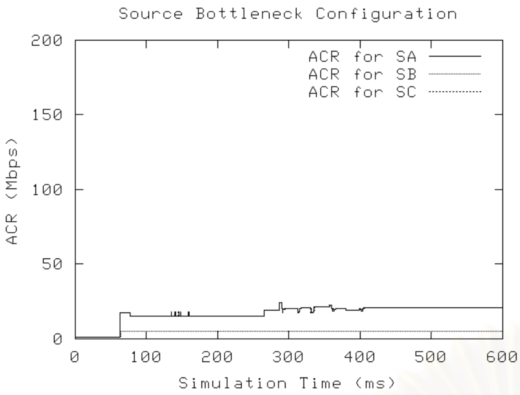


(c) Queue Length

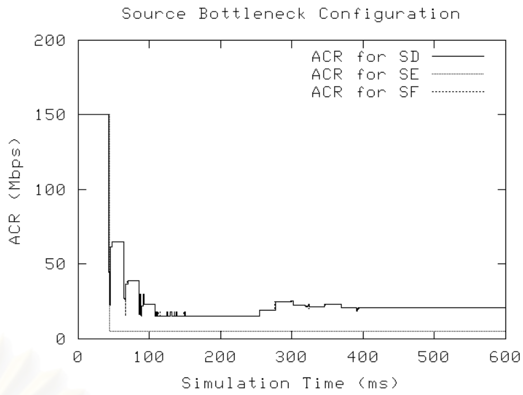


(d) Link Utilization

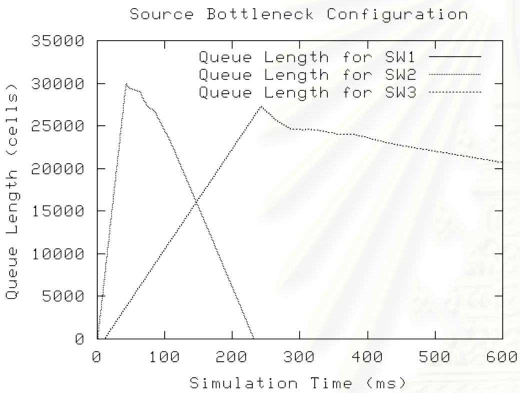
Figure 4.14 Simulation results of source bottleneck configuration with R-S algorithm



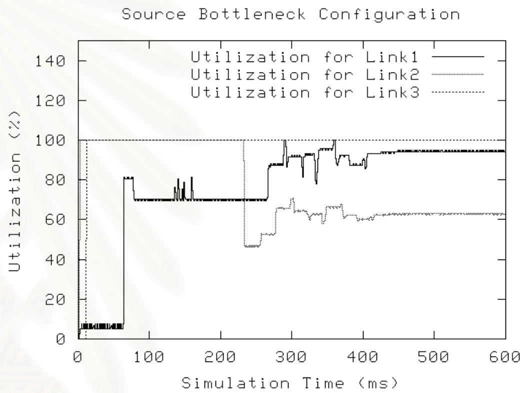
(a) ACR



(b) ACR

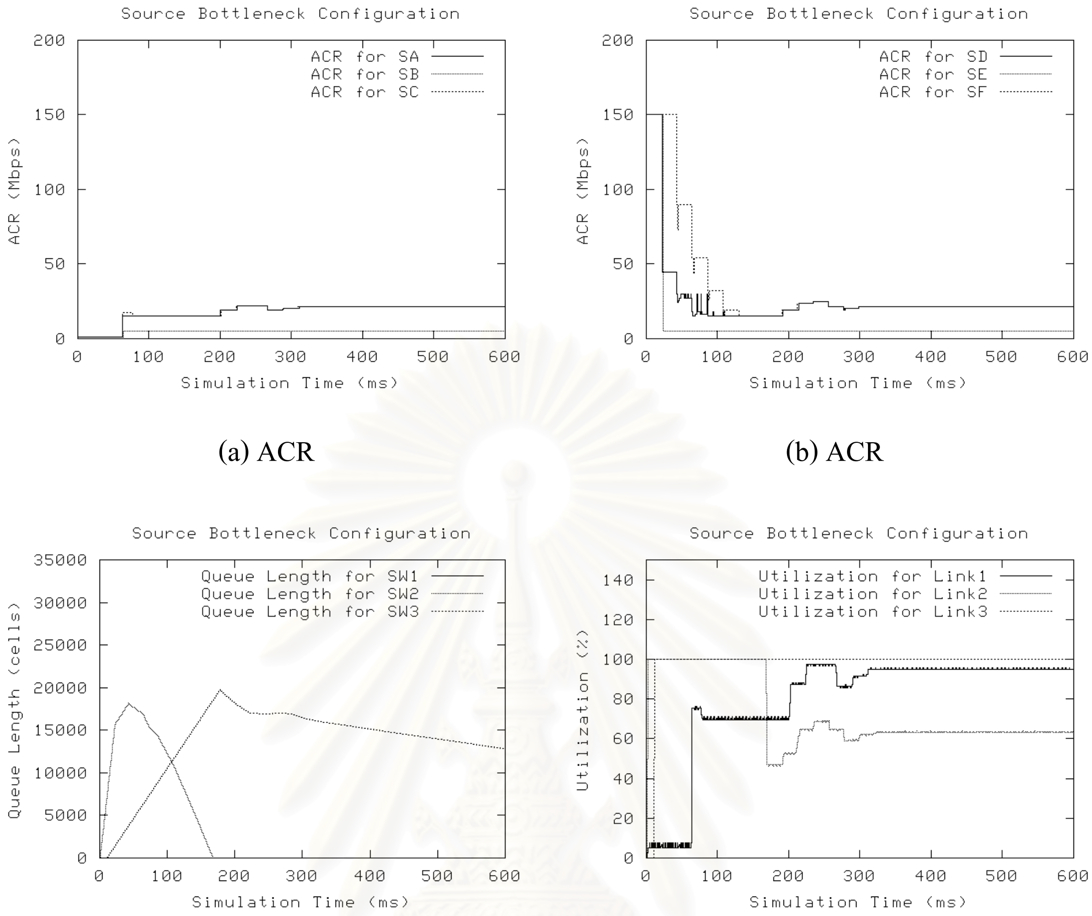


(c) Queue Length



(d) Link Utilization

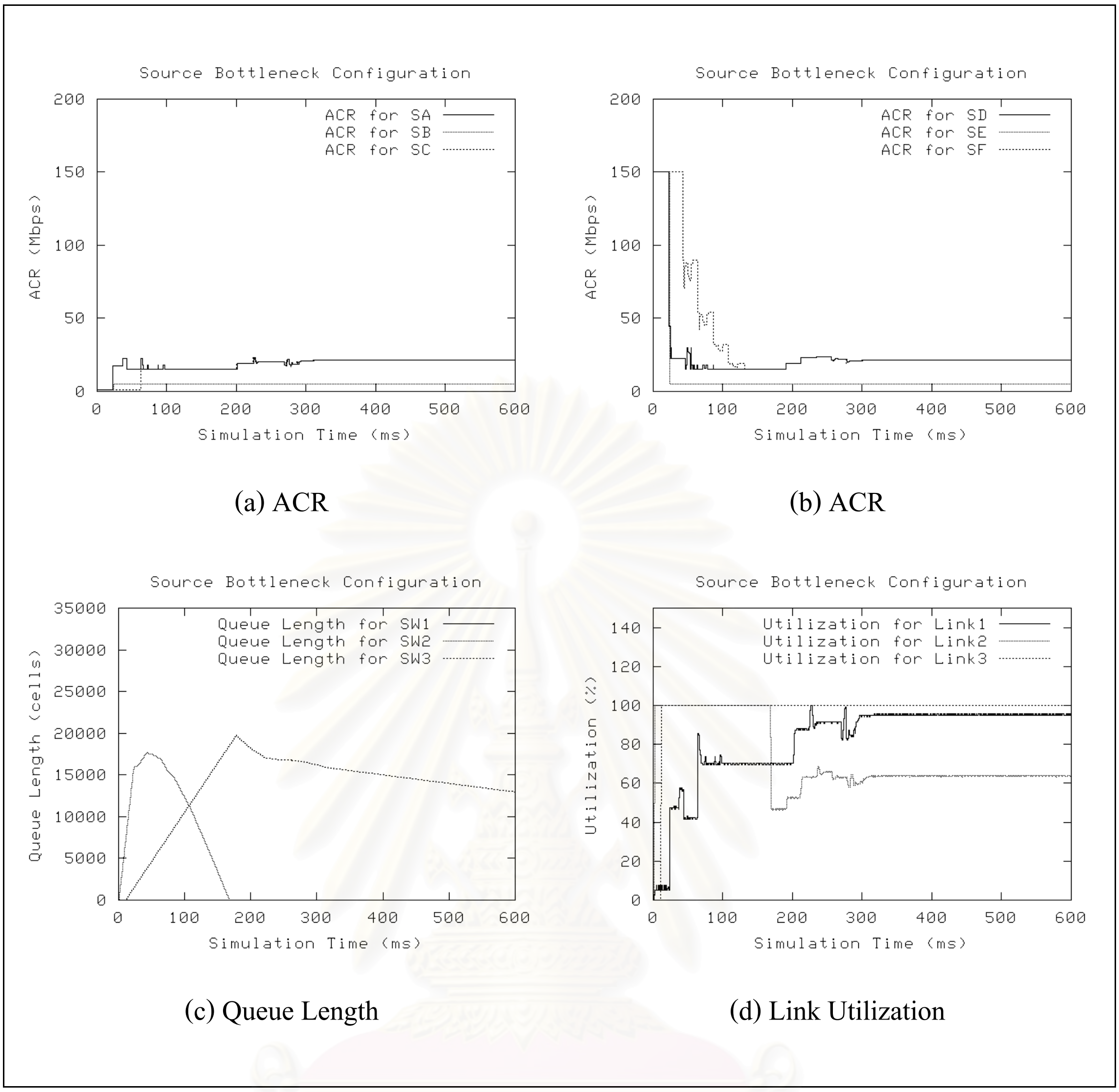
Figure 4.15 Simulation results of source bottleneck configuration with Ren algorithm



(a) ACR (b) ACR

(c) Queue Length (d) Link Utilization

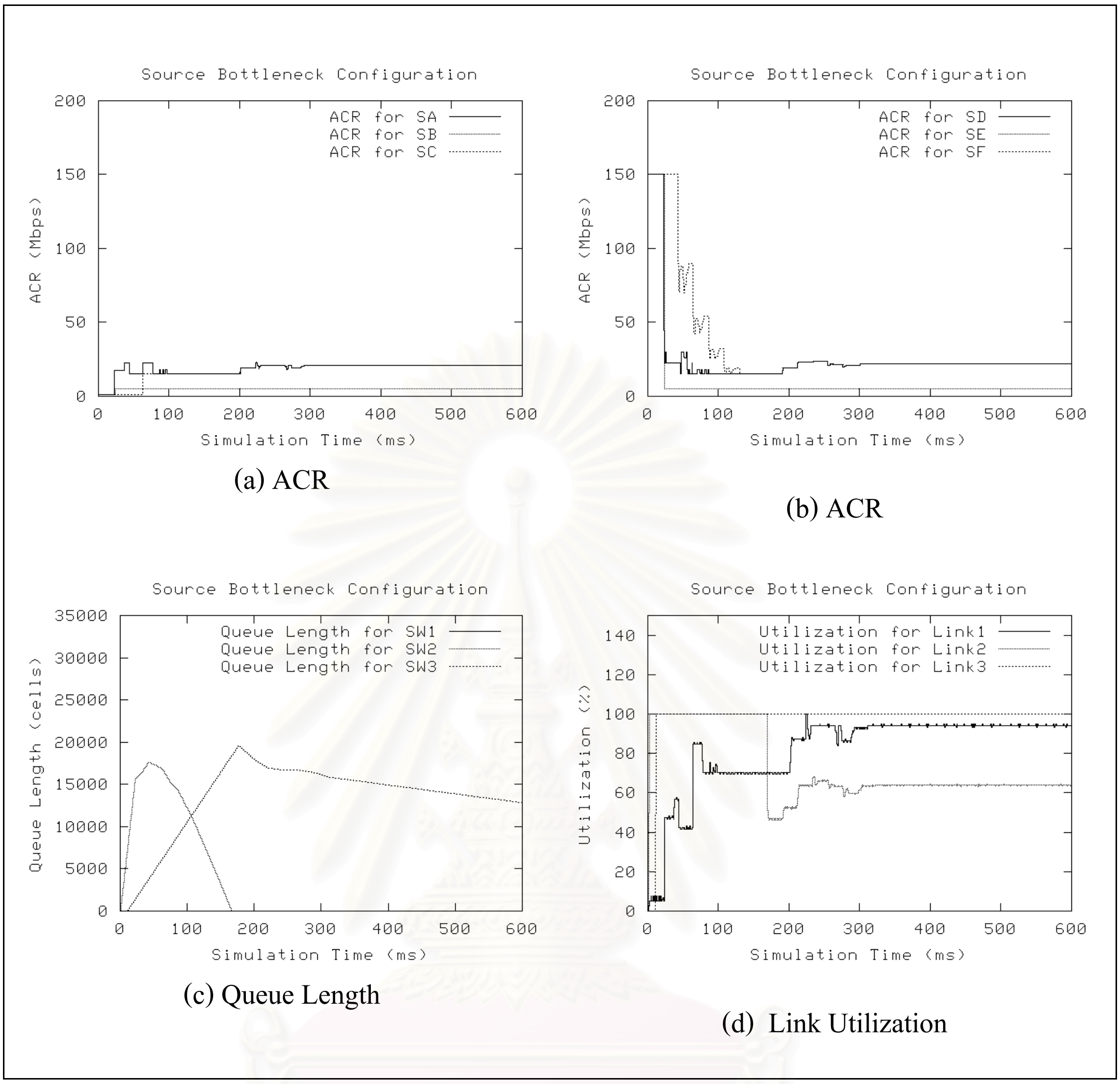
Figure 4.16 Simulation results of source bottleneck configuration with Fahmy algorithm



(a) ACR (b) ACR

(c) Queue Length (d) Link Utilization

Figure 4.17 Simulation results of source bottleneck configuration with RQB algorithm



(a) ACR

(b) ACR

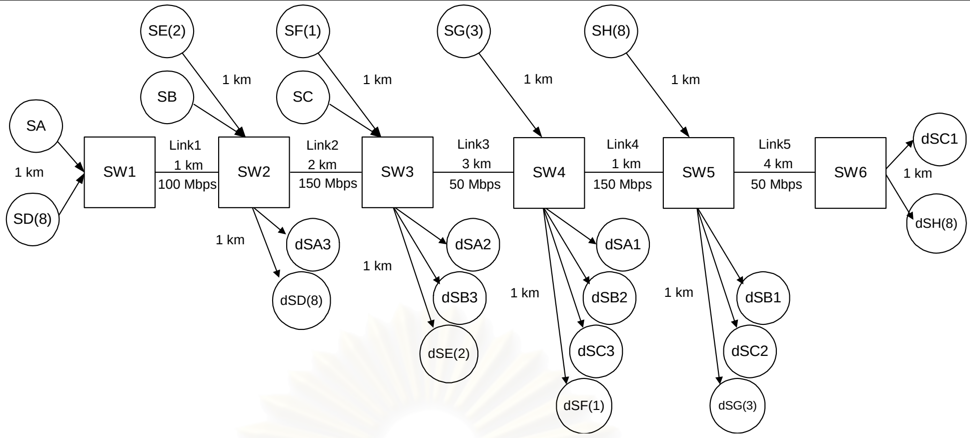
(c) Queue Length

(d) Link Utilization

Figure 4.18 Simulation results of source bottleneck configuration with SBF algorithm

From the simulation results shown in Figure 4.14 to Figure 4.18, every algorithm can allocate the bandwidth to every sources according to the Max-Min fairness scheme. The significant differences are the ACR oscillation for the R-S algorithm and the queue length at SW2 and SW3 for the Ren algorithm. In Figure 4.14, the ACR of the source fluctuates due to the characteristic of the algorithm. This leads to a higher degree of link utilization fluctuation compare to other algorithms. Figure 4.15c shows the queue length at SW2 and SW3 of Ren algorithm, it is noticeably higher than that of Fahmy's in Figure 4.16c. The reason is that Fahmy algorithm can detect an overload condition at SW2 then it immediately controls the data rate of SD and SF to the lower value to relieve the congestion at the switch without waiting for all BRM cells.

4.3.4 LAN Configuration



LAN Configuration
(Fairness and Transient Response Test in LAN topology)
Link1 is the congested link for SA and SD(8)
Link3 is the congested link for SB, SC and SF(1)
Link5 is the congested link for SC and SH(8)
Propagation delay: 5 us/1 km
All switch-to-end system links are 1 km
Target Utilization = 0.90 (90%), RIF = 1, AI = 1 ms

Figure 4.19 LAN configuration

We choose this model to test the fairness and transient response of the consolidation algorithms for the low propagation delay networks. The system under test consists of 3 multicast sources which are SA, SB and SC and 22 unicast sources which compose of SD(8), SE(2), SF(1), SG(3) and SH(8). From Max-Min fairness bandwidth calculation, the bandwidth that each source should be allocated is as follows:

- SA and SD(8) congest at Link 1, therefore, each source will get ((100*0.9)/9) = 10 Mbps.
- SC and SH(8) congest at Link 5, therefore, each source will get ((50*0.9)/9) = 5 Mbps.
- SB and SF(1) will get the bandwidth which is left from SA and SC occupying on Link 3 equal to ((50*0.9)-(10+5))/2 or 15 Mbps.
- SE(2) can use Link 2 bandwidth which is left over from SA and SB. That is equal to ((150*0.9)-(10+15))/2 or 55 Mbps.
- SG(3) will get the bandwidth which is left from the occupying of SB and SC on Link 4 as equal to ((150*0.9)-(15+5))/3 or 38.33 Mbps.

Figure 4.20 to Figure 4.24 are the simulation results of the network using Not wait-for-all, Wait-for-all, Fahmy, RQB and SBF algorithm, respectively. All of them exhibit an insignificant difference because of a short link distance in the network. Hence, the effect due to the propagation delay of the slow response algorithms is not obviously seen in LAN environment.

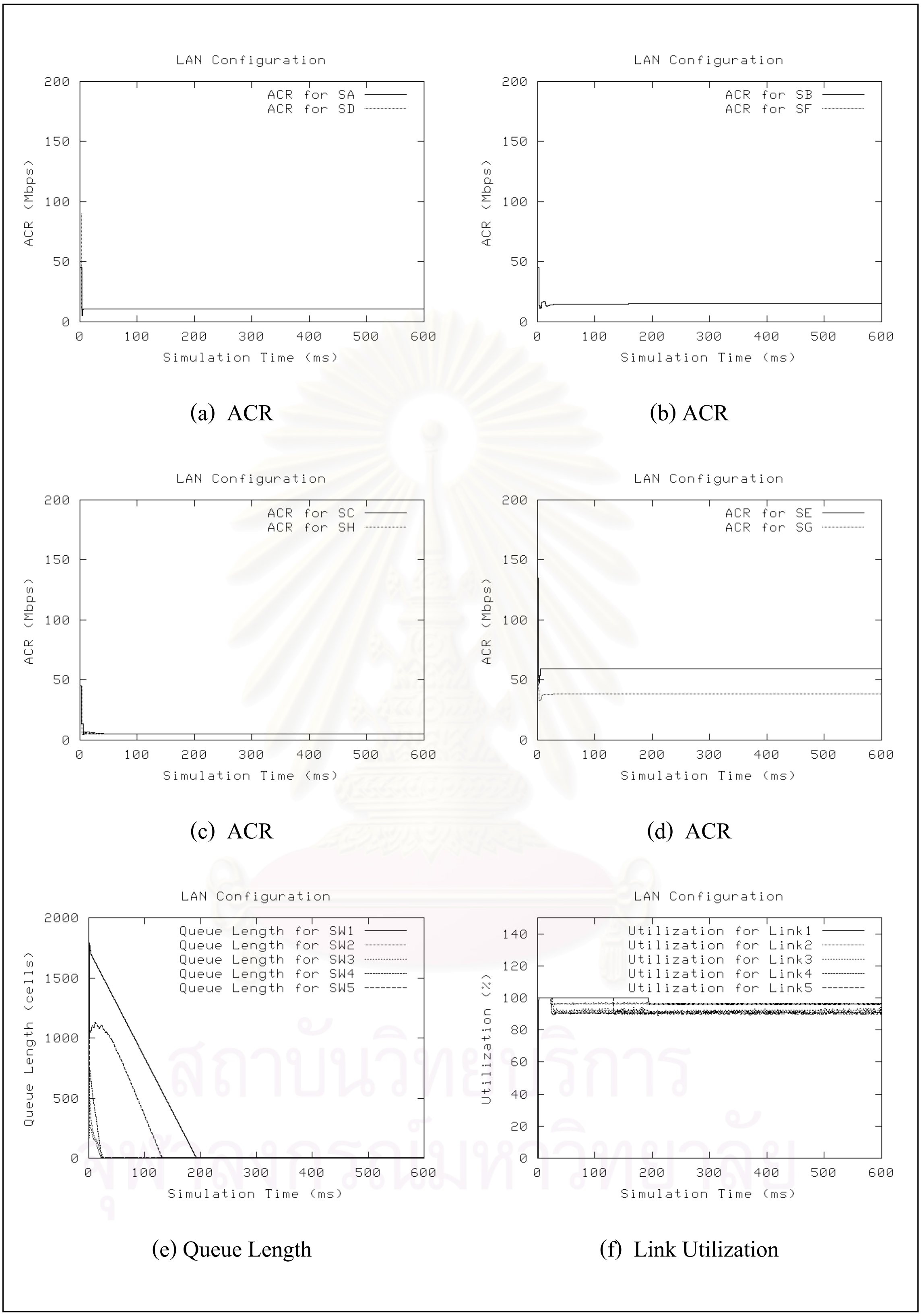


Figure 4.20 Simulation results of LAN configuration with R-S algorithm

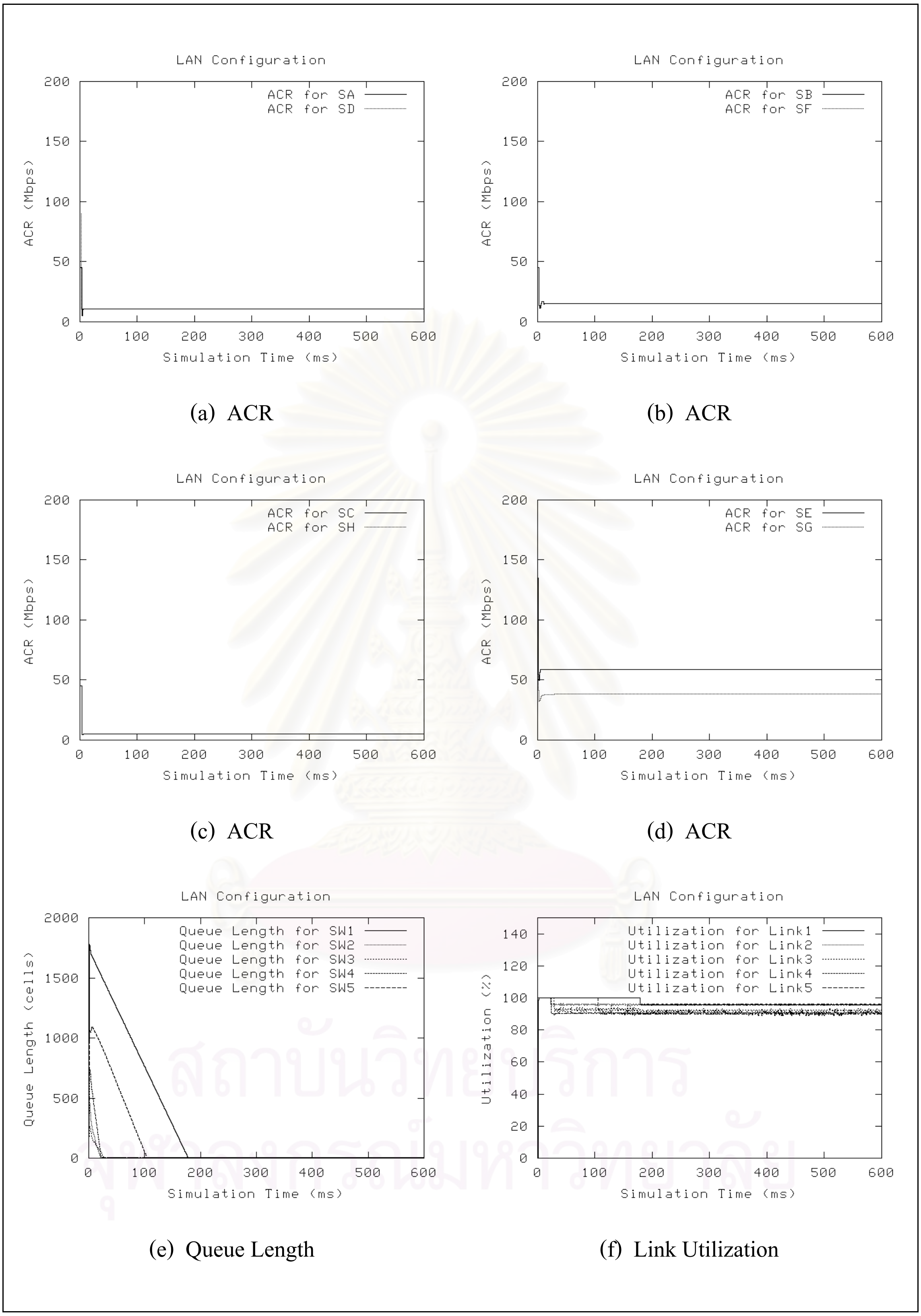
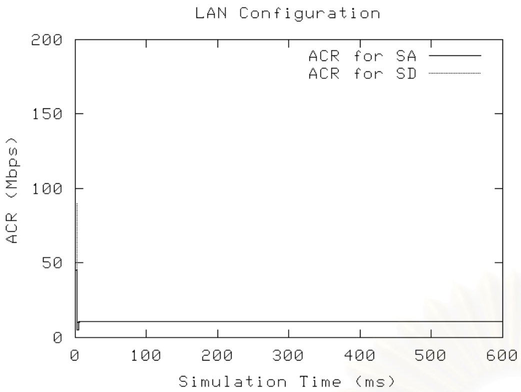
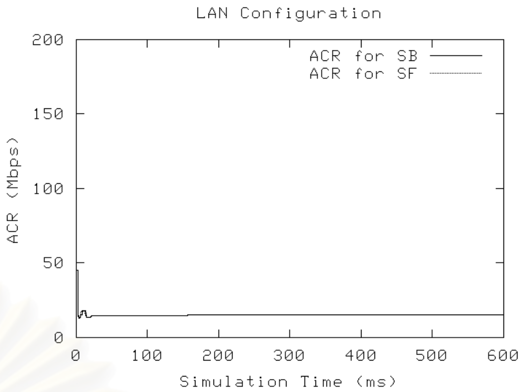


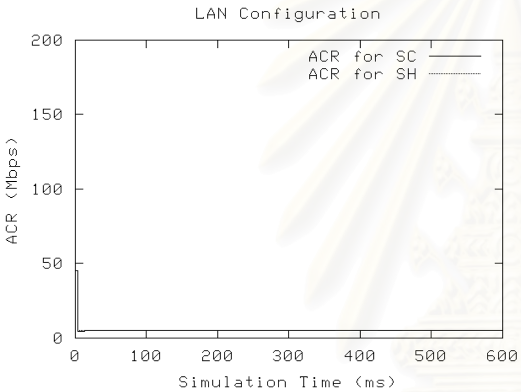
Figure 4.21 Simulation results of LAN configuration with Ren algorithm



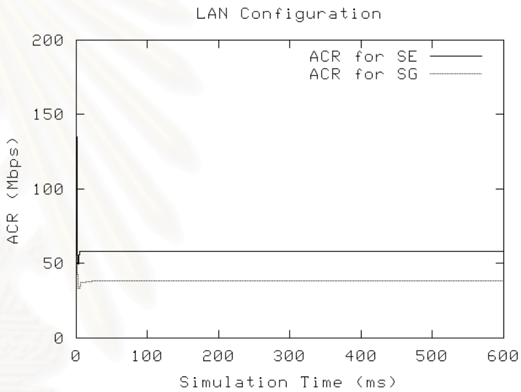
(a) ACR



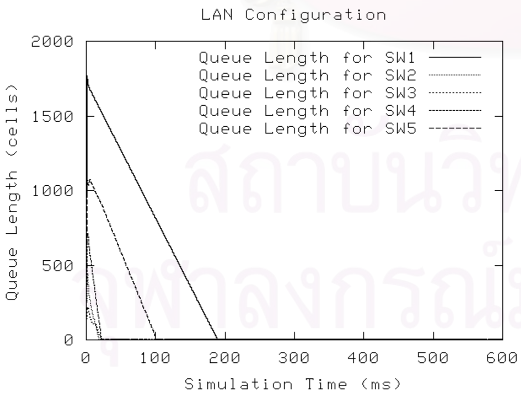
(b) ACR



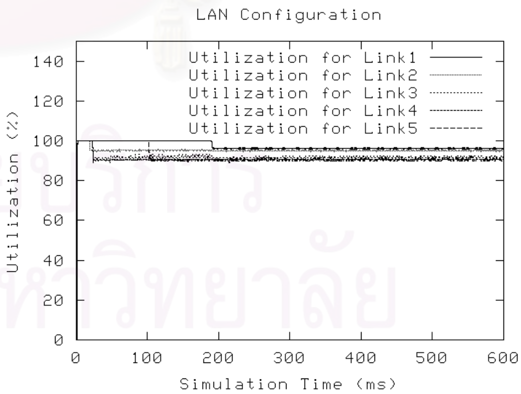
(c) ACR



(d) ACR

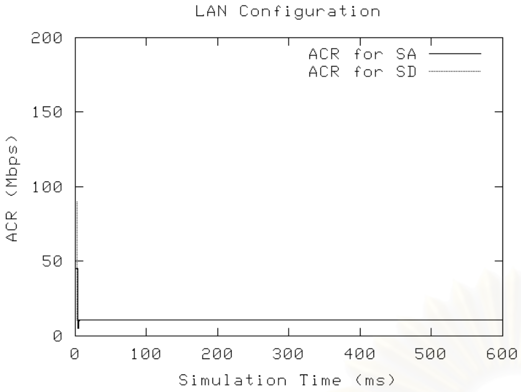


(e) Queue Length

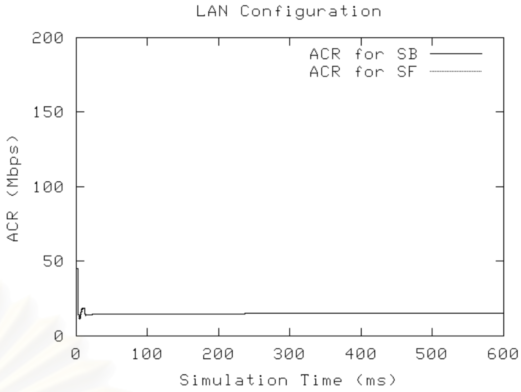


(f) Link Utilization

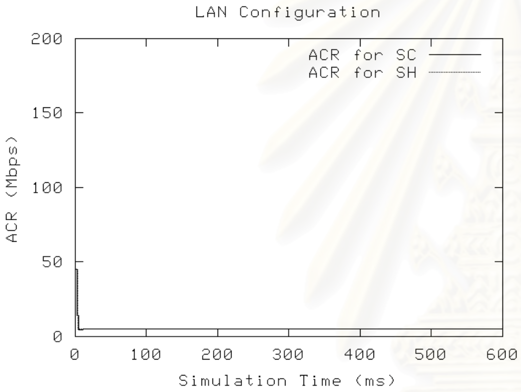
Figure 4.22 Simulation results of LAN configuration with Fahmy algorithm



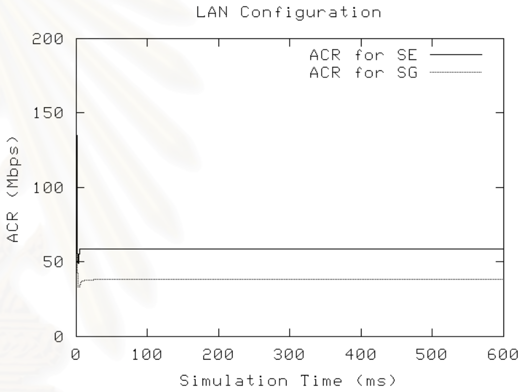
(a) ACR



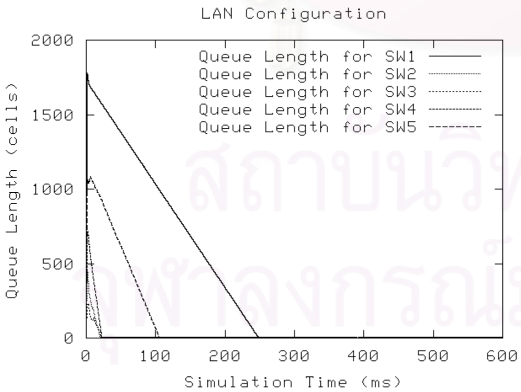
(b) ACR



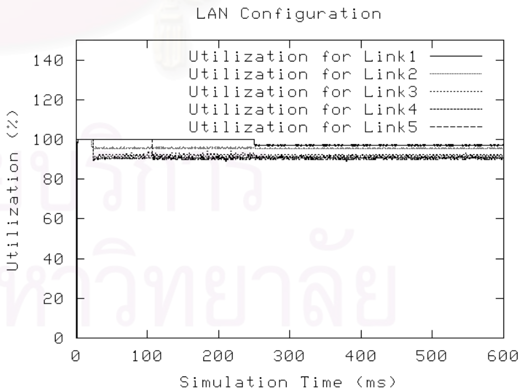
(c) ACR



(d) ACR

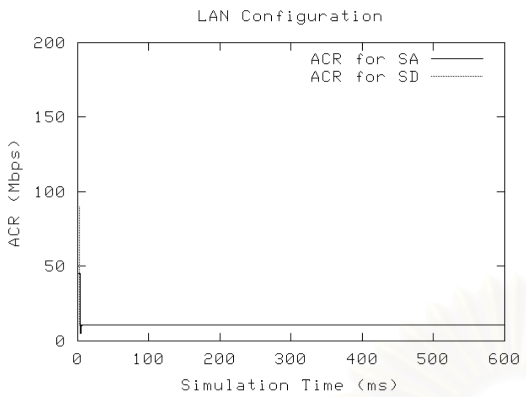


(e) Queue Length

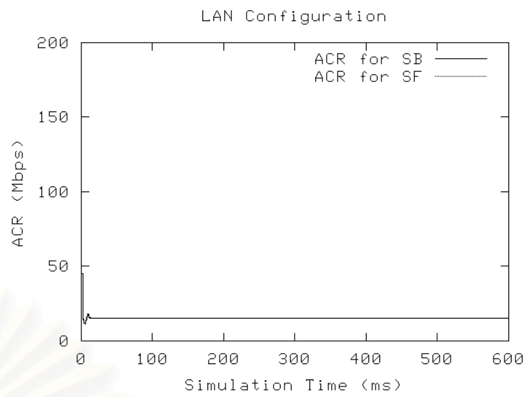


(f) Link Utilization

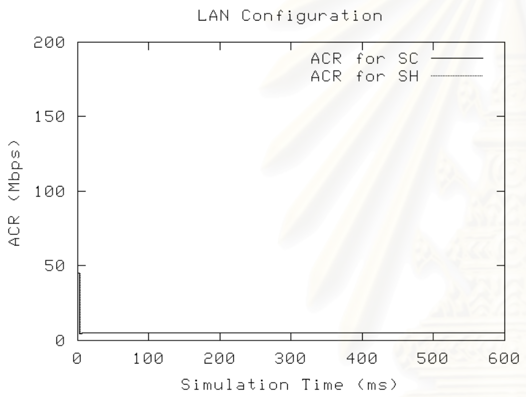
Figure 4.23 Simulation results of LAN configuration with RQB algorithm



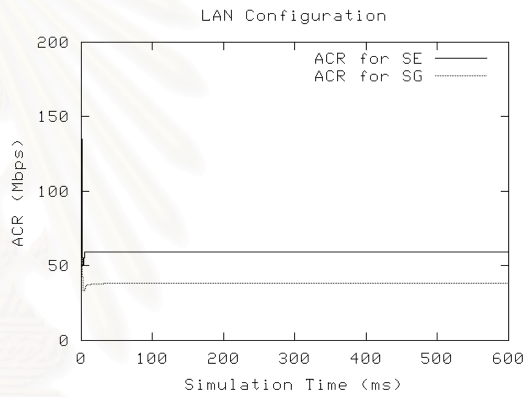
(a) ACR



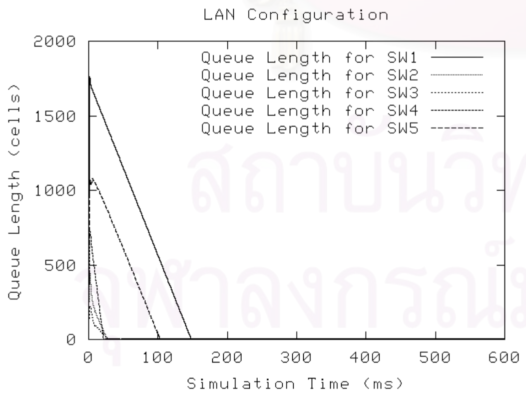
(b) ACR



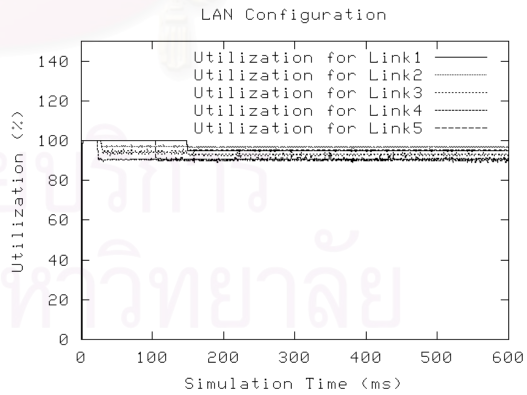
(c) ACR



(d) ACR



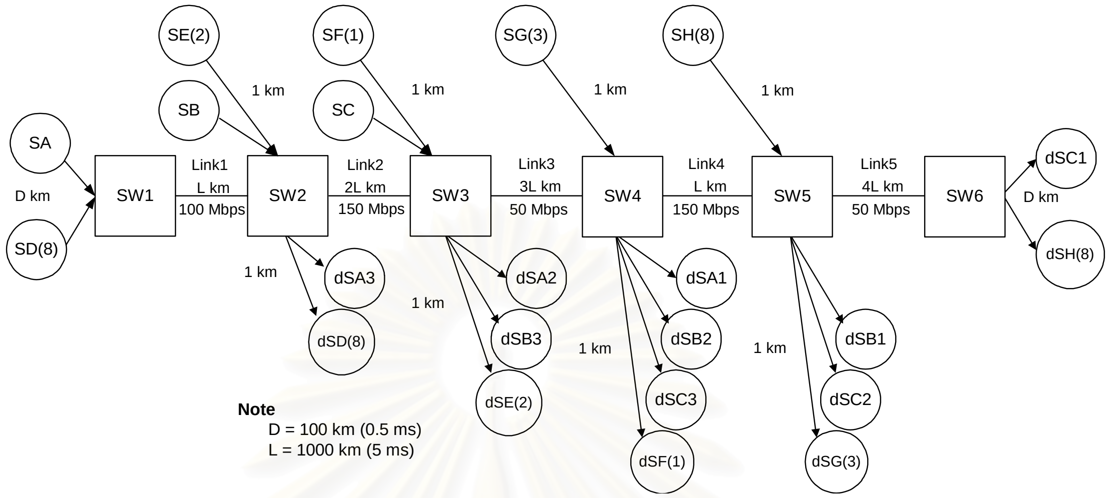
(e) Queue Length



(f) Link Utilization

Figure 4.24 Simulation results of LAN configuration with SBF algorithm

4.3.5 WAN Configuration

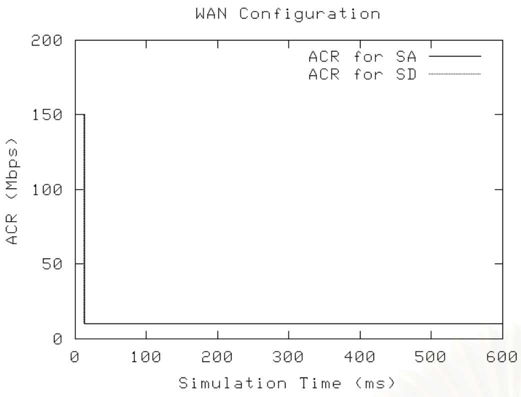


<u>WAN Configuration</u>

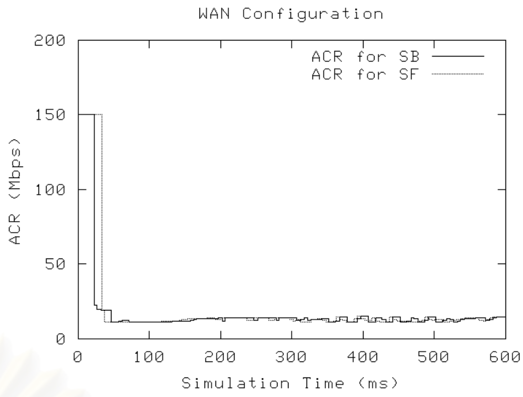
(Fairness and Transient Response Test in WAN topology)
Link1 is the congested link for SA and SD(8)
Link3 is the congested link for SB, SC and SF(1)
Link5 is the congested link for SC and SH(8)
Distance D = 100 km (0.5 ms) and Distance L = 1000 km (5 ms)
All switch-to-end system links are 1 km
Target Utilization = 0.90 (90%), RIF = 1, AI = 1 ms

Figure 4.25 WAN configuration

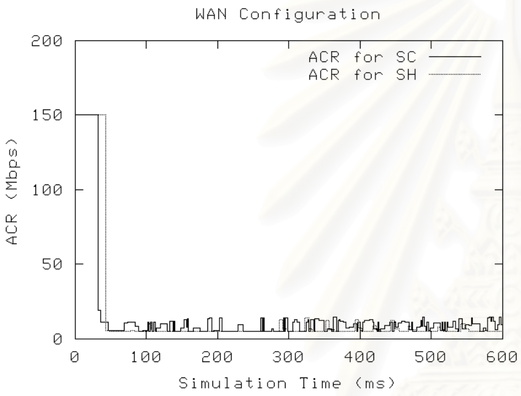
This configuration has the same topology as in LAN except the distance of inter-switch links. We extend the distance of the link to study the effect of the propagation delay to each consolidation algorithm. As the network topology is the same as in LAN, therefore, the bandwidth of each source is also similar to that calculated in LAN case. That is 10 Mbps, 15 Mbps, 5 Mbps, 10 Mbps, 55 Mbps, 38.33 Mbps and 5 Mbps for SA, SB, SC, SD, SE, SF, SG and SH, respectively. The simulation results in Figure 4.26 to Figure 4.30 show that all algorithms can adjust the source rate to the calculated value. The difference is the transient response or the time to get to fair share of the source. SA, SD, SB and SF of wait-for-all algorithm take almost three times longer than that of the rest algorithms. For SC, SH, SE and SG, wait-for-all algorithm takes nearly 100 ms to get into a steady state while the other algorithms consume only half time of it. Queue length is the consequence of the transient response. The slow rate adjustment of the source in wait-for-all algorithm causes the data continuously injected to the network at the rate that is higher than the network can support (ICR value is set to PCR, one can argue why we do not set the ICR to the value lower than PCR. We can, but will suffer from the low link utilization). Notice that the average queue length of wait-for-all algorithm is higher than that of the others.



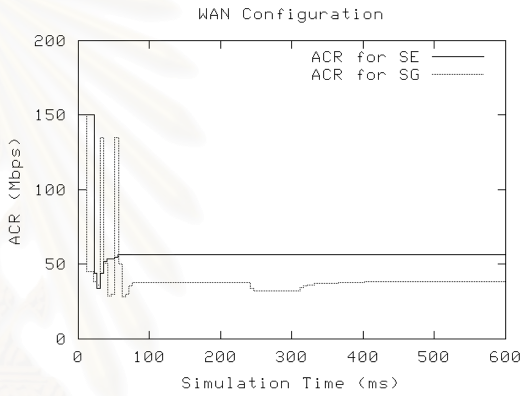
(a) ACR



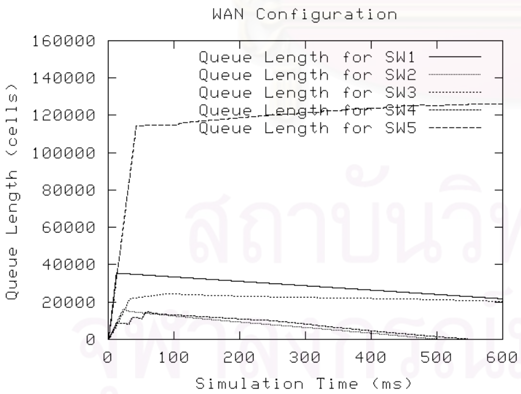
(b) ACR



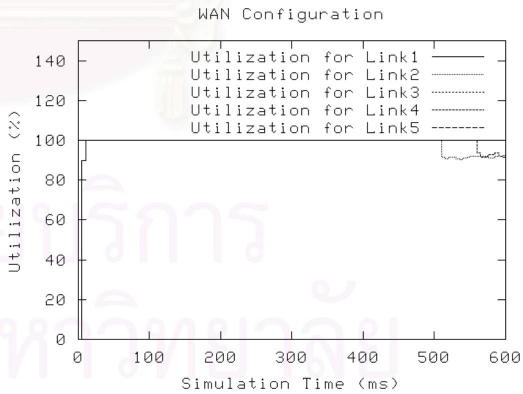
(c) ACR



(d) ACR

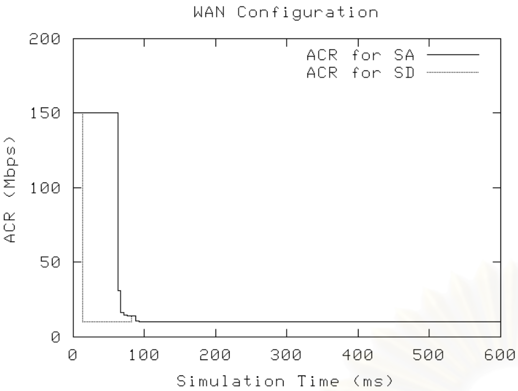


(e) Queue Length

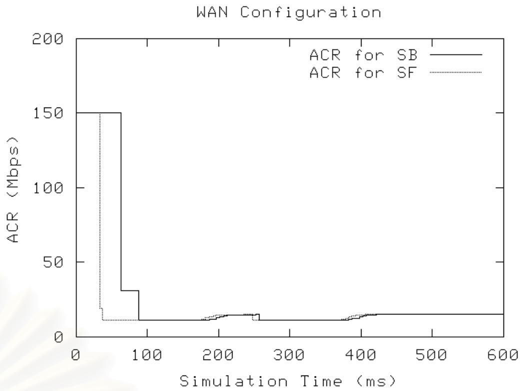


(f) Link Utilization

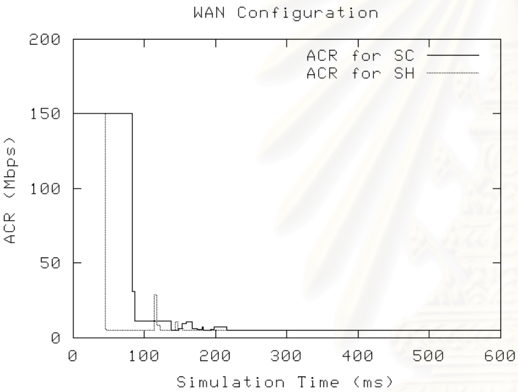
Figure 4.26 Simulation results of WAN configuration with R-S algorithm



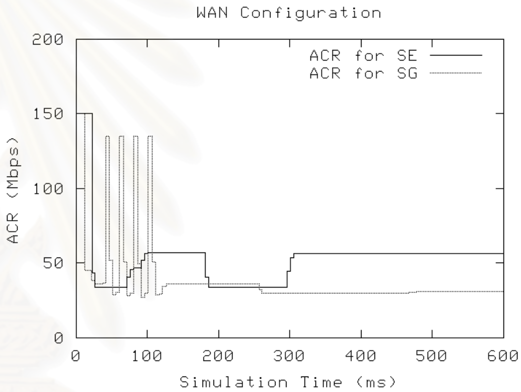
(a) ACR



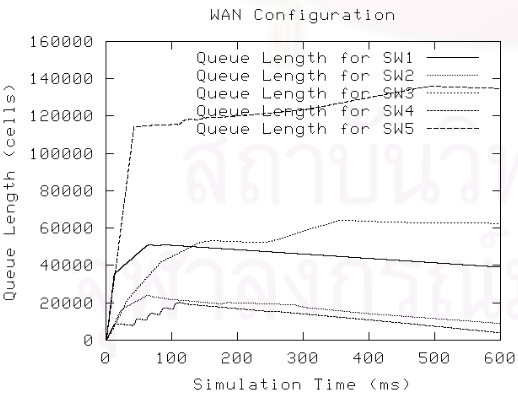
(b) ACR



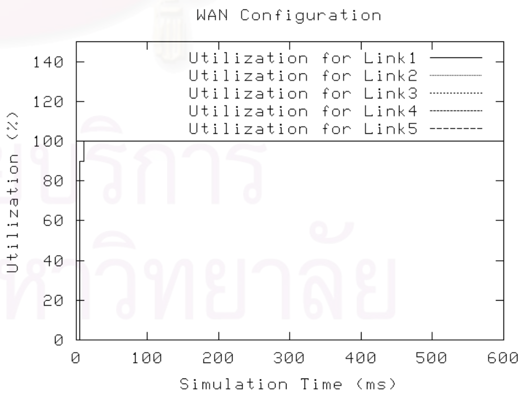
(c) ACR



(d) ACR

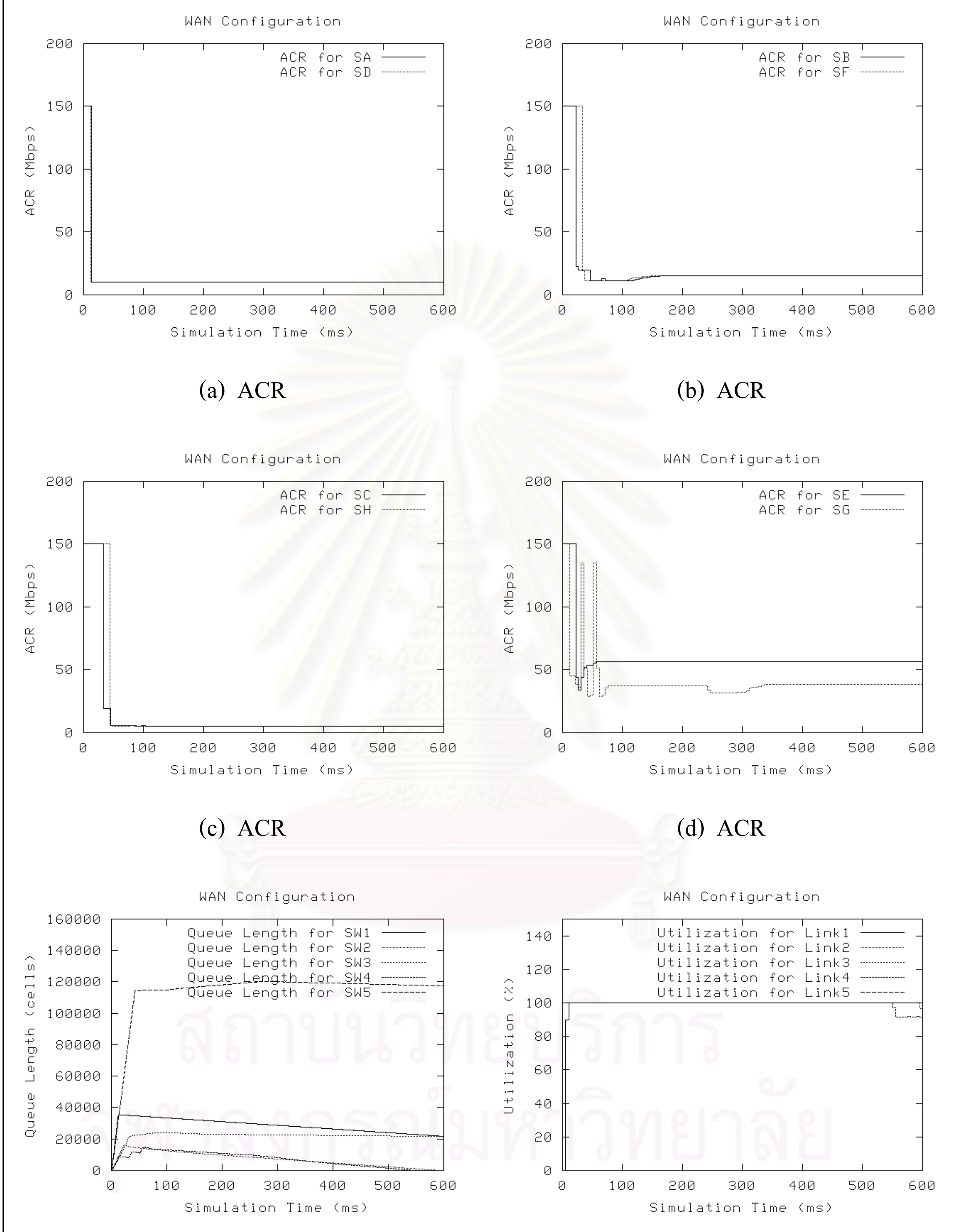


(e) Queue Length



(f) Link Utilization

Figure 4.27 Simulation results of WAN configuration with Ren algorithm

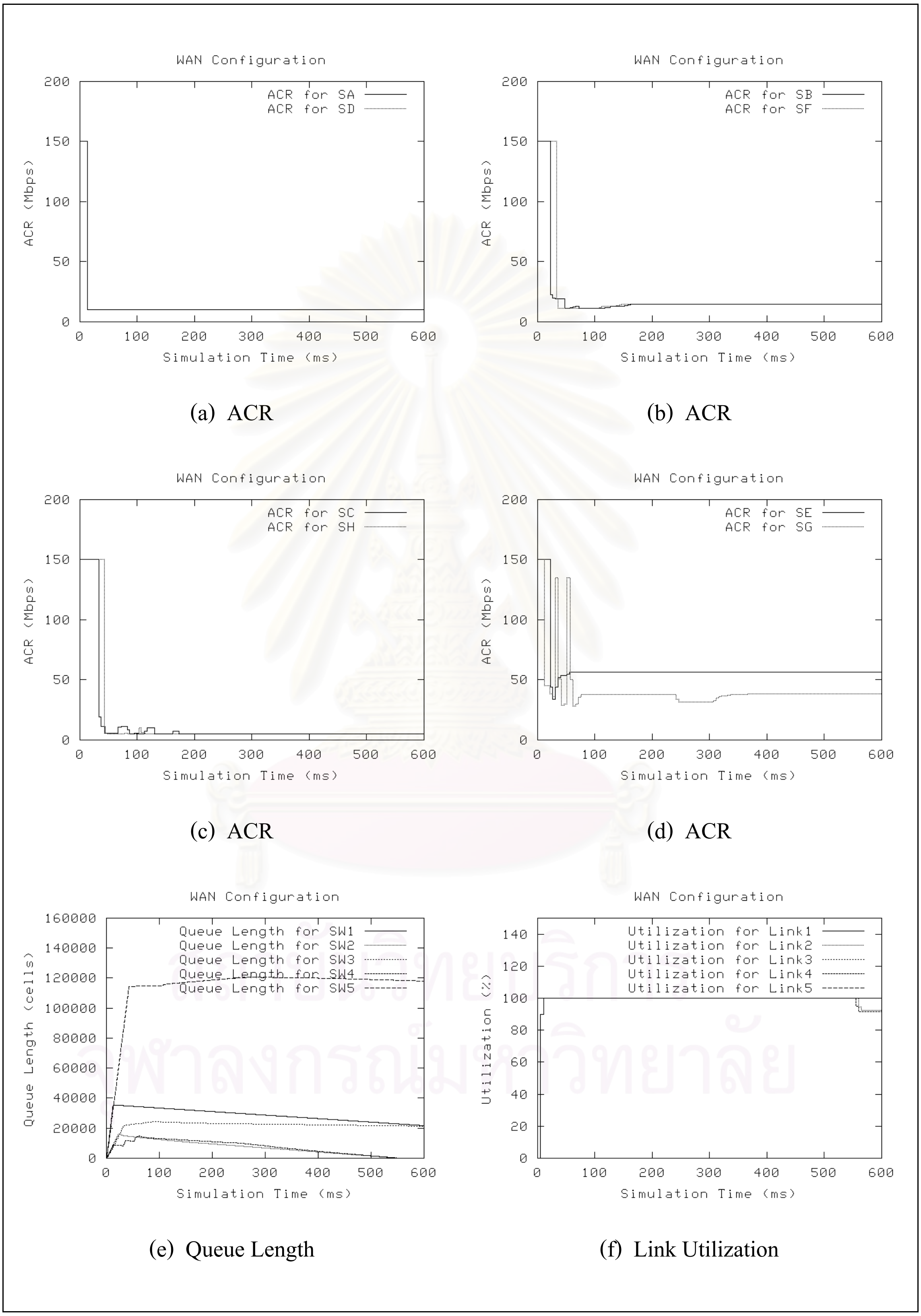


(a) ACR (b) ACR

(c) ACR (d) ACR

(e) Queue Length (f) Link Utilization

Figure 4.28 Simulation results of WAN configuration with Fahmy algorithm

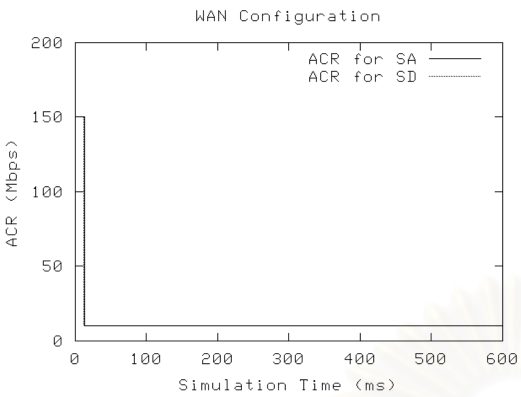


(a) ACR (b) ACR

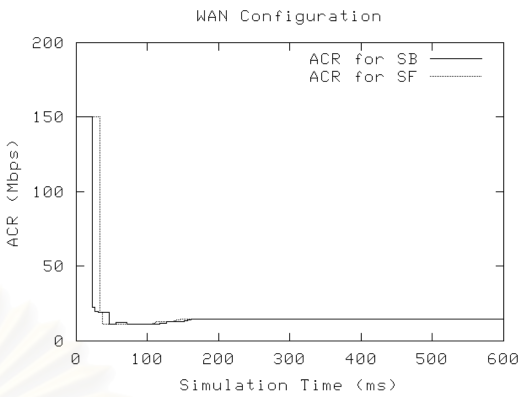
(c) ACR (d) ACR

(e) Queue Length (f) Link Utilization

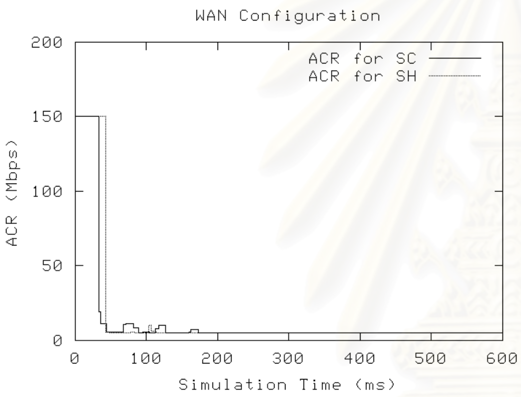
Figure 4.29 Simulation results of WAN configuration with RQB algorithm



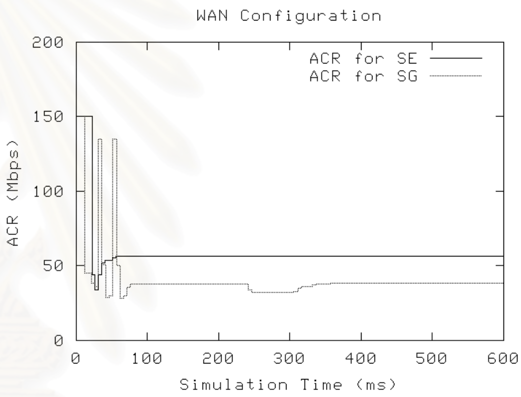
(a) ACR



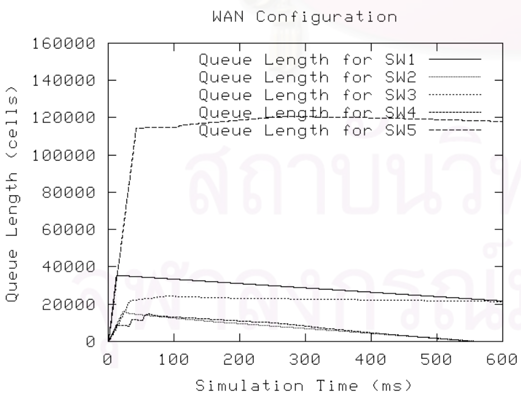
(b) ACR



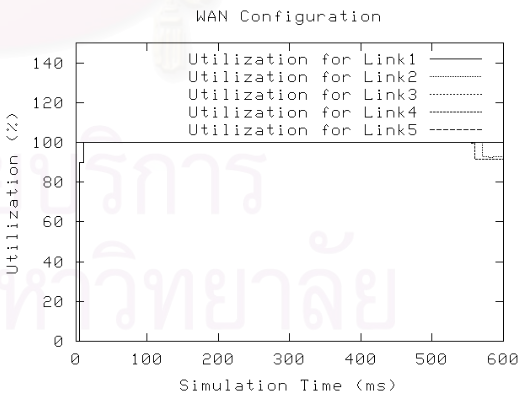
(c) ACR



(d) ACR



(e) Queue Length



(f) Link Utilization

Figure 4.30 Simulation results of WAN configuration with SBF algorithm

The performance comparison of each consolidation algorithm can be made from the WAN network configuration in Figure 4.5. We choose this configuration because of its various in source types and the burstiness of the traffic in the network. We compare many performance aspects e.g. transient response, consolidation noise and etc. of each algorithm by using the results from simulation. The value transient response in Table 4.1 is the transient response time of S1.

The performance comparison of our proposed and the other consolidation algorithms is provided in Table 4.1

Table 4.1 Performance comparison between existing and proposed consolidation algorithms

Algorithm	*Wait-for-all (Ren)*	*Not wait-for-all (R-S)*	*Fahmy*	*RQB*	*SBF*
Complexity	Low	High	Low	Low	Low
Transient Response	41	6.5	6.5	6.5	6.5
Consolidation Noise	Low	High	Low	Low	Low
BRM to FRM ratio at root	≤ 1	≤ 1	Converge to 1	> 1	Converge to 1
Scalability (Robust to network expansion)	No	Yes	No	No	Yes

The complexity of the algorithms is compared based on the way of generating BRM cell at the branch point. For all algorithms except not-wait-for-all algorithm, although the number of registers and flag bits used in each algorithm are different. However, the complexity of those algorithms is considerable at the same level because the number of register in a processing unit is not a constraint with today's solid state technology. Hence, these algorithms can be considered as a low complexity algorithm. Notice for not-wait-for-all, its 'high' complexity is caused from the BRM cell generated by the branch point in stead of receiving from the destination as other algorithm. The BRM cell-generating process consumes major part of processing power due to it has to create a whole new cell, calculate the appropriate rate, fill in the ER field and modify some other parameters. Most studies [35], [58]-[60] state that generating RM cell has a high implementation cost. The rest of algorithms do not generate RM cell and in this sense the complexity for wait-for all algorithm should be higher than others. Because the simulation program used in this thesis is unable to trace down to the processing time level, hence complexity of algorithms cannot be illustrated quantitatively.

The transient response and consolidation noise shown in the table is obvious from the simulation results. Scalability of consolidation algorithm is the property of the algorithm that the consolidation delay and propagation delay will not increase with the number of branch points in the multicast session. The wait-for-all does not pose this property because its response time relates directly to the network size. For Fahmy and RQB, basically they function like wait-for-all algorithm but they have a feature to change their operation according to the network condition (not always work in a wait-for-all mode). Although the sensitivity to number of branch point should be better than that of wait-for-all, however they should be classified that they are lack of scalability. For not-wait-for-all and SBF, due to their operation, which feedback the BRM cell from the nearest branch without waiting for the others. Subsequently, they will not be affected by the network expansion.

CHAPTER 5

Analysis of Response Time and Source Rate for Consolidation Algorithms

This Chapter is devoted to study the response time and source rate of consolidation algorithms. We focus on Wait-for-all, Not-wait-for-all, Fahmy and SBF algorithm [44]. For RQB algorithm, although it outperforms Wait-for-all and Not-wait-for-all algorithm, we omit to analyze the equation because it does not fully comply with the ATM Forum Traffic Management Version 4.0 Specification (By using of Queue Length field in RM cell) [42].

Given a multi-level branch point network model in Figure 5.1, the response time and ACR are analyzed and presented mathematically as follows.

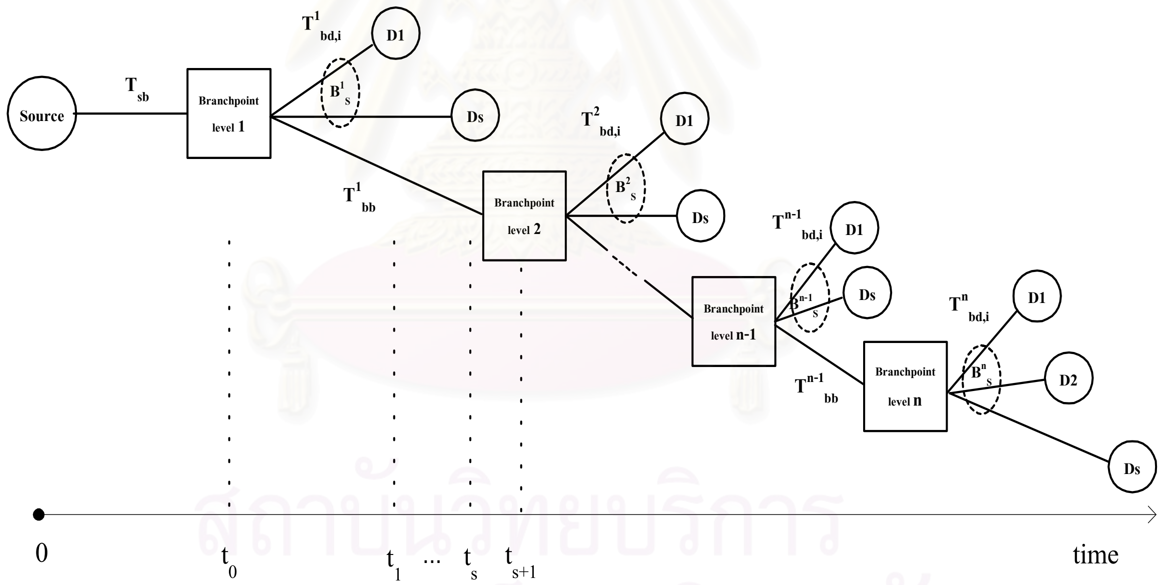


Figure 5.1 Network model for response time and source rate approximation

Let $t_0 = 2T_{sb}$

$t_1 = 2T_{sb} + 2T^1_{bd,i}$ *and so on.*

$t_0 < t_1 < ... < t_s$

t_{s+1} may be greater or less than t_s

Notations

$s \in S$ denotes a multicast session

B^n_s denotes the set of branches connected between branch point and destinations of s at level n

n is the number of branch point level

T_{sb} is a propagation time from the source to branch point

$T^n_{bd,i}$ is a propagation time from branch point to destination via branch i at level n

T^n_{bb} is a propagation time from branch point level n to level n+1

T^k is a time that branch point level n takes to response to branch point level k

T_r is a response time of the source to the network

T_{rW}, T_{rNW}, T_{rFahmy} and T_{rSBF} are the response time of Wait-for-all, Not wait-for-all, Fahmy and SBF algorithm

$t_{RTDmax.}$ denotes the maximum round trip delay for the source to destination

$r_i(t)$ denotes the ER value corresponding to branch i computed by branch point at time t

$b_{s,i}(t)$ denotes the value of ER in BRM cell of session s received by branch point via branch i at time t

Definition *A response time of the consolidation algorithm in point to multipoint connection is the time duration counting from the source send the first cell out until it gets the first BRM cell back.*

5.1 Response Time

Case I: Wait-for-all Algorithm

As described in section 3.4.1 of chapter 3, the branch point has to wait for BRM from all branches. In a single branch point level case (*n*=1), there are only destinations connecting to the branch point. Assume that the processing time and the queuing delay time is neglected, therefore, the time that branch point takes to consolidate the BRM cells from all branches can be expressed as

$$T^1 = \max_{i \in B_s^1}(T^1_{bd,i}) \tag{5.1}$$

According to Figure 5.1 and the algorithm described in section 3.4.1, the response time of a wait-for-all algorithm for single level branch point can be formed as follows.

$$T_{rW} = 2(T_{sb} + \max_{i \in B_s^1}(T^1_{bd,i})) \tag{5.2}$$

In the case that there is more than one branch point level ($n \geq 2$ levels) in the network, the response time is tend to be higher. For the sake of simplicity, we consider response of the network from the most downstream level up to the level 1. The equation is turned to be a recursive function and the response time is function of number of level as shown below.

$$T^{n} = \max_{i \in B_s^n}(T_{bd,i}^{n}) \tag{5.3}$$

$$T^{n-1} = max(\max_{i \in B_s^{n-1}}(T_{bd,i}^{n-1}),(T_{bb}^{n-1} + \max_{i \in B_s^n}(T_{bd,i}^{n})))$$

$$T^{n-2} = max(\max_{i \in B_s^{n-2}}(T_{bd,i}^{n-2}),(T_{bb}^{n-2} + T^{n-1}))$$

$$\vdots$$

$$T^{k} = max(\max_{i \in B_s^{k}}(T_{bd,i}^{k}),(T_{bb}^{k} + T^{k+1})) \quad ; k = n-1, n-2, ..., 1 \tag{5.4}$$

Consider at level n (the most downstream one), the response time eq.(5.3*)* of this level is similar to that of single level case. We use this equation as an initial value to find the response time at level 1in eq.(5.4). Subsequently, the response time of wait-for-all algorithm in the network can be expressed by

$$T_{rW} = 2(T_{sb} + T^{k}) \tag{5.5}$$

Case II: Not wait-for-all Algorithm

For the not wait-for-all algorithm described in section 3.4.2 of chapter 3, the branch point sends a BRM cell back upon the reception of FRM when at least one BRM has been received from a destination. If we neglect the inter-FRM cell time, the time that the first BRM arrives at the branch point is the shortest branch that connected to the branch point. The equation of this time is formulated in eq.(5.6)

$$T^{1} = min(\min_{i \in B_s^1}(T_{bd,i}^{1}), T_{bb}^{1}) \tag{5.6}$$

where $T_{bb}^{1} = \infty$ when the network has only one branch point level. In other words, this implies that in a single level branch point, T_{bb}^{1} does not exist. The response time of the not wait-for-all can be expressed by

$$T_{rNW} = 2(T_{sb} + T^{1}) \qquad (5.7)$$

We can see that the response time of the not wait-for-all is determined by the shortest path connecting to the branch point and it is independent of the number of branch point level.

Case III: Fahmy Algorithm

For Fahmy algorithm, described in section 3.4.3 of chapter 3, it works basically on a wait-for-all basis. The difference is that it has a capability of detecting an overload condition at the branch point. When the overload is detected, the branch point will not wait for any in-arrival BRMs but it sends the existing BRM immediately to the source instead. The algorithm now works as a not wait-for-all mode. We cannot analyze the exact value of the response time for this algorithm because the randomness of overload occurrence. It may be equal to or less than that of wait-for-all. Hence, the response time of Fahmy can be bounded as illustrated in eq.(5.8).

$$T_{rW} \geq T_{Fahmy} \geq T_{rNW} \qquad (5.8)$$

Case IV: SBF Algorithm

For SBF algorithm described in section 4.2 of chapter 4, since the algorithm works on a not wait-for-all basis but it can send the BRM back to the source without waiting for the arriving of FRM. Hence, the response time is less than or equal to that of not wait-for-all algorithm. That is

$$T_{rSBF} = T_{rNW} - \frac{n}{ICR} \quad ; 0 \leq n \leq 31 \qquad (5.9)$$

where $\frac{1}{ICR}$ equal to the time between consecutive ATM cell running at the speed of ICR cell per second and n is the number of cells arrived at branch point since the last arriving of BRM cell. In case that n equal to zero, it means that the transient response of SBF is equal to that of Not-wait-for-all. So far, we have analyzed the response time of the many types of consolidation algorithms. It can be concluded that the wait-for-all poses a slowest response time. Fahmy algorithm which works on a wait-for-all basis but has a superior in detecting a congestion at the branch point exhibit a faster response. SBF and the not wait-for-all algorithm offer the best service in terms of response time. From equation (5.8) and (5.9) we can conclude that

$$T_{rW} \geq T_{rFahmy} \geq T_{rNW} \geq T_{rSBF} \qquad (5.10)$$

5.2 Allowed Cell Rate

Case I: Wait-for-all Algorithm

Since the branch point has to wait for BRM cells from all branches before sending BRM cell to the source. Hence, only one BRM cell per source has been sent during the $\left|0, t_{RTD\max.}\right|$ interval. We define the possible source rate during any time interval as $R(t_{start}, t_{end})$ where t_{start} and t_{end} is the starting and the ending time of the observed interval, respectively. According to Figure 5.1 and algorithm described in section 3.4.1, the source rate of wait-for-all algorithm can be formulated as in eq.(5.11).

$$\begin{aligned} &R(0, t^{-}_{RTD\max}) = ICR \\ &R(t_{RTDmax.}, t^{++}_{RTDmax.}) = \left|min\left(b_{s,i}(t), r_i(t)\right)\right| \quad ; \forall i \in B^1_{s+1} \end{aligned} \qquad (5.11)$$

where *ICR* is an Initial Cell Rate of the source and $B^n{}_{s+1}$ is $B^n{}_s$ includes the link between the branch point level n and level n+1.

Notice that the ratio of number of BRM cells sent from branch point to source over the number of FRM cells received by the branch point, $\frac{BRM}{FRM}$, is less than one and this will make the algorithm to be less adaptive to the changing traffic.

Case II: Not wait-for-all Algorithm

For not wait-for-all algorithm, since branch point sends a BRM cell to the source on the reception of a FRM cell when at least one BRM cell has been received from a destination. If there are many BRM cells arriving at the branch point during two consecutive FRMs, the branch point will send only one BRM cell which contains the minimum ER value among those BRM cells and the ER value that are computed by the branch point. After sending the BRM cell to the source, the MER register at the branch point is set to PCR value. This is to allow the source to generate the cell rate upto PCR value. Otherwise, the source rate will get lower and lower at each time the BRM has been sent. However, this may be suffered from the consolidation noise. Assume that $\left|t_i, t_{i+1}\right| > \left|\;\right|_{FRM}$ where $\left|t_i, t_{i+1}\right|$ and $\left|\;\right|_{FRM}$ is the difference of propagation time between destination i and i+1 to the source (see Figure 5.1) and the time interval of two consecutive FRM cells, respectively. The $\left|\;\right|_{FRM}$ interval is not a fixed value. It can be varied with the source cell rate instead. If the source cell rate increases the $\left|\;\right|_{FRM}$ decreases. Normally, it is set to have a 31 cell intervals i.e. a source may send 32 cells for each FRM cell. Let *j* be the number of BRM cells sent by branch point to source. Hence, there will be $j = \left\lceil \frac{\left|t_i, t_{i+1}\right|}{\left|\;\right|_{FRM}} \right\rceil$ cells sent by the branch point to control the source rate during $\left|t_i, t_{i+1}\right|$ interval. Referring to Figure 5.1, then we can analyze the source rate as

$$R(0,t_0^-) = ICR \tag{5.12}$$

$$\begin{aligned}
&R(t_0,t_1^-) = \left|min\left(ICR, r_i(t)\right)\right|_{FRM,1\,to\,j} \qquad ; \forall i \in B_{s+1}^{1} \\
&R(t_1,t_2^-) = \left|min\left(PCR, b_{s,1}(t), r_i(t)\right)\right|_{FRM,1\,to\,j} \quad ; \forall i \in B_{s+1}^{1} \\
&\quad \vdots \\
&R(t_i,t_{i+1}^-) = \left|min\left(PCR, b_{s,k}(t), r_i(t)\right)\right|_{FRM,1\,to\,j} \\
&\qquad\qquad\qquad ; k = 1,2,\ldots,i \quad ; \forall i \in B_{s+1}^{1} \\
&R(t_{RTDmax.}, t_{RTDmax.}^{++}) = \left|min\left(PCR, b_{s,i}(t), r_i(t)\right)\right|_{FRM,1\,to\,j} \\
&\qquad\qquad\qquad\qquad ; \forall i \in B_{s+1}^{1}
\end{aligned} \tag{5.13}$$

Since, $b_{s,i}(t)$ is the value of ER in BRM cells generated from the network. Its value depends on the parameters setting in the network elements, the network topology, the traffic condition etc. Hence, it can be varied with time for each inter-FRM interval. The inter-FRM interval itself is varied according to the change of source rate. If the next FRM interval is wider (lower source rate) than the current one the branch point has more time to collect the BRM cells and the source rate tends to have less or even no oscillation. Oppositely, in case that the FRM interval is narrower (at $t_{RTDmax.}$) than the branch point that has less time to collect the BRM cells and will cause the rate to fluctuate between $b_{s,i}(t)$ and $r_i(t)$. The consolidation noise is produced at this time and lasts forever because the FRM and BRM lose their synchronism in sending and receiving cells for each FRM interval.

Case III: Fahmy Algorithm

For Fahmy algorithm, as described before, if there is no overload at the branch point it works in a wait-for-all mode. The source rate equation looks similar eq.(5.11). In case an overloaded condition is detected at the branch point, the branch point will send instantly a BRM with the minimal value between ER in BRM at that time and the calculated fair share value to the source. After that it becomes to work as before i.e. wait-for-all and does not response to the source regardless how the bandwidth of the network will change. The ACR of Fahmy can be formulated as follows.

$$R(0,t_{Overload}^{-}) = ICR \tag{5.14}$$

$$R(t_{Overload},t_{RTDmax.}^{-}) = \left|min\left(b_{s,i}(t_{Overload}),r_i(t)\right)\right| \quad ; \forall i \in B_{s+1}^{1} \tag{5.15}$$

$$R(t_{RTDmax.},t_{RTDmax.}^{++}) = \left|min\left(b_{s,i}(t),r_i(t)\right)\right| \quad ; \forall i \in B_{s+1}^{1} \tag{5.16}$$

where $t_{Overload}$ is the time when an overloaded condition is detected.

Case IV: SBF Algorithm

For SBF algorithm, the branch point receives the BRM from the branches and may or may not update the ER and Branch_Number depending on where these two parameters fall in which of the 4 given cases in the algorithm. After updating the ER and Branch_number information, BRM is sent back to the source independently without waiting for the arriving of FRM cell. Consequently, during the transient state the source rate can be adapted very fast and accordingly to the available bandwidth in the network. During the steady state, since the algorithm always send the least $b_{s,i}$ value to the source there is no noise introduced no matter how the FRM interval will change. The source rate of SBF can be formulated as follows.

$$
\begin{aligned}
&R(0,t_0^{-}) = ICR \\
&R(t_0,t_1^{-}) = \left|min\left(ICR,r_i(t)\right)\right| \quad ; \forall i \in B_{s+1}^{1} \\
&R(t_1,t_2^{-}) = \left|min\left(ICR,b_{s,1}(t),r_i(t)\right)\right|_{BRM} \; ; \forall i \in B_{s+1}^{1} \\
&\quad\vdots \\
&R(t_i,t_{i+1}^{-}) = \left|min\left(b_{s,k}(t),r_i(t)\right)\right|_{BRM} \\
&\qquad\qquad ; k = 1,2,...,i \;\; ; \forall i \in B_{s+1}^{1}
\end{aligned}
\tag{5.17}
$$

$$R(t_{RTDmax.}, t^{++}_{RTDmax.}) = \left| min\left(b_{s,i}(t), r_i(t)\right) \right|_{BRM} \quad ; \forall i \in B^{1}_{s+1} \tag{5.18}$$

where $\left| \quad \right|_{BRM}$ is the time interval of any two consecutive arriving BRM cells from any branches in the network.

CHAPTER 6

Interoperation of Consolidation Algorithms

In this chapter, we investigate the interoperability of the consolidation algorithm. We focus on the response time and ACR problems that may occur in the branch points interoperated networks. In the networks with many branch points implemented with different consolidation algorithms, the characteristics of each consolidation algorithm will affect the others according to the algorithms running at branch point A and B. To my best knowledge, this work is a very first one that proposes the interoperation issue and the derivation of the response time and ACR of various consolidation algorithms [45]. In addition, we will investigate which branch point in the network will play a major role in determining the network performance.

6.1 Performance Evaluation

In a heterogeneous consolidation algorithm network, each branch point has a different response time depending on the characteristic of the algorithm implementing on it. The question is which branch point will be the most influential one. How the characteristic of the consolidation algorithm running at branch point A and B in Figure 6.1 and Figure 6.7 plays a role in determining the network performance. To answer this, we have to know the response time and ACR of each algorithm. Fortunately, we have already analyzed them mathematically in the previous chapter and will use them to explain the results that will be carried out in this chapter. We use the response time, ACR and asymmetrical round trip delay as performance metrics. The simulation will be carried out extensively to evaluate the network performance in both long round trip delay (WAN) and short round trip delay (LAN/MAN) cases.

6.2 Simulation Results and Discussions

Parameters Setting

- *Except where indicated, all link capacities are 150 Mbps.*
- *All switch-to-end system links are 50 kms except where specified.*
- *All sources are persistent sources.*
- *The source parameter Rate Increase Factor (RIF) is set to 1.*

- *The Peak Cell Rate (PCR) is set to 150 Mbps. The Initial Cell Rate (ICR) is set to 150 Mbps*
- *Use ERICA switch algorithm with Max-Min fairness as a rate allocation algorithm.*
- *The switch target utilization is set to 90%.*
- *Averaging Interval (AI Time) is 0.1ms.*

6.2.1 WAN Configuration

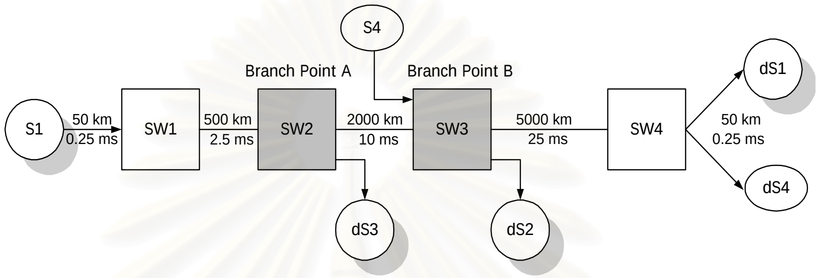


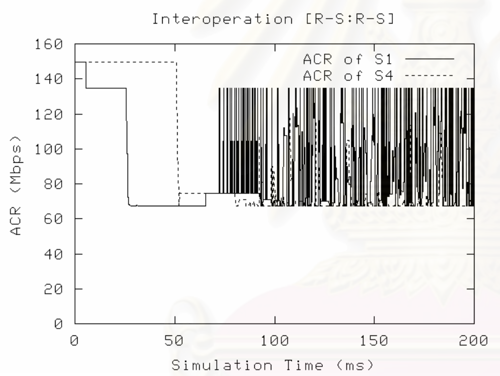
Figure 6.1 Network model I

In WAN configuration, the link distances between the branch points were selected to emphasize the effect of the transient response and consolidation noise and source's ACR under the long propagation delay environment. Besides the multicast session (S1, dS1, dS2, dS3 and dS4), we have also put a unicast session (S4 and dS4) in the network to verify the fairness of bandwidth sharing of the interoperated algorithms.

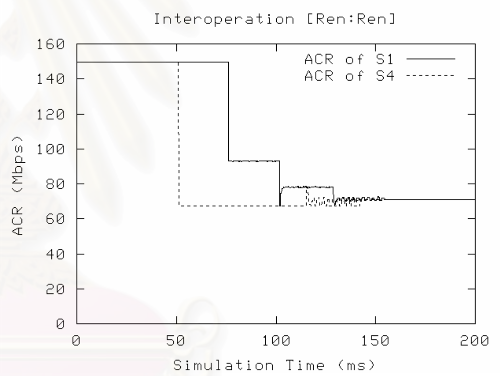
In order to show that the algorithm performs well even in the severe condition, the parameters were set to an extreme case. For example, RIF is set to one (RIF range from 1/32768 to 1 [12]) to allow the source to generate cells at the full explicit rate indicated in the returning RM cells. ICR is also set to a very high value (equal to PCR). These parameters were selected to emulate the worst case load situation.

The switch target utilization and AI time are parameters of the ERICA rate allocation algorithm. The target utilization is a parameter which is typically set to 90% of the available capacity. AI time is an interval that the load is measured and averaged by the switch. We set AI to 0.1 ms to ensure that the switch will receive at least one cell for averaging during the interval (the cell interval for 150 Mbps link speed is 2.83 μs). On the other hand, if the AI is set to a too-large value compared to the propagation delay in the network, the switch traffic load measurement will not be performed frequently enough.

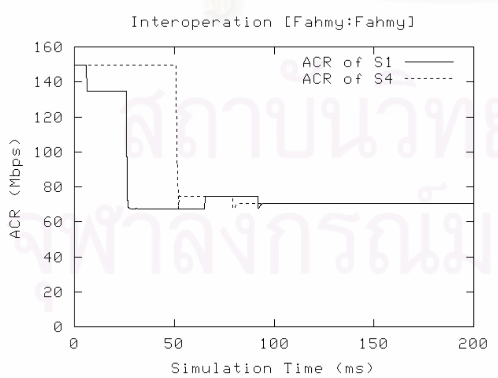
We investigate the interoperation by using both the same consolidation algorithm and different consolidation algorithms in the branch points. The Network Model I representing a WAN configuration is shown in Figure 6.1. S1 is a source for a point-to-multipoint connection and its destinations are dS1, dS2 and dS3. A round trip delay from S1 to dS1, dS2 and dS3 are 76 ms, 26 ms and 6 ms, respectively. S4 is a source for point-to-point connection and the round trip delay to its destination, dS4, is 51 ms. SW1 and SW4 are the switching nodes and SW2 and SW3 are the branch points. The algorithms which have been described in chapter 3 and chapter 4 are implemented in branch point A (SW2) and branch point B (SW3). Branch point A is considered as an upper stream branch point while branch point B is a lower stream branch point. For the sake of simplicity we will use to notation e.g. [R-S:Ren] to represent the R-S and Ren algorithms used at branch point A and branch point B, respectively. S1 and S4 are set to initially start sending with ICR value. This setting causes a high network utilization while exhibits a congestion and large queue length at branch point B. To avoid the congestion, S1 and S4 should reduce their ACR to the fair share rate as fast as they can. However, ACR decreasing could not be occurred before a source has received a turn-around RM cell from its destination.



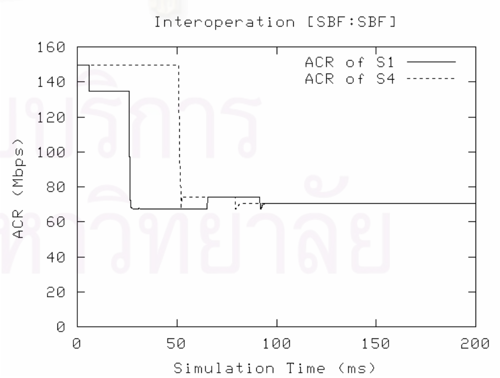
(a)



(b)

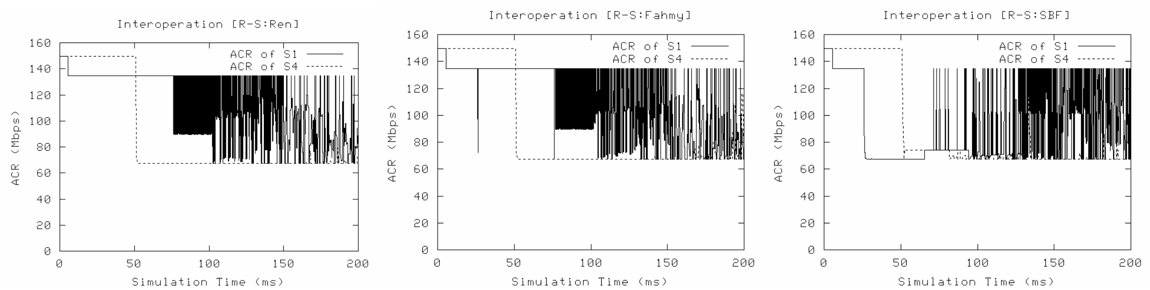


(c)



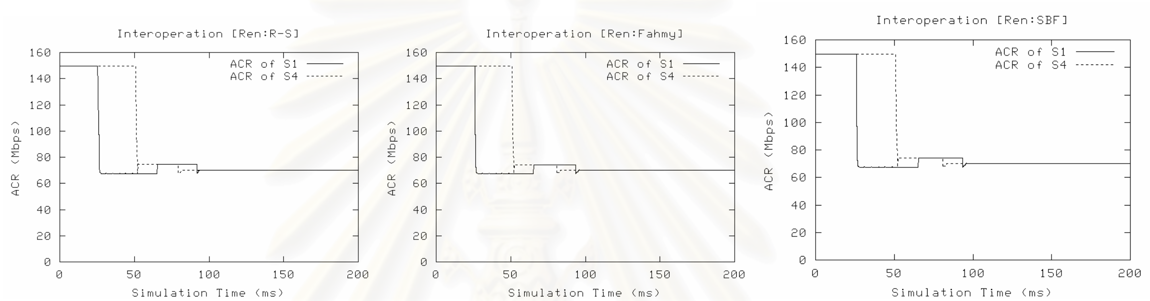
(d)

Figure 6.2 Simulation results of WAN configuration using the same consolidation algorithm at branch points



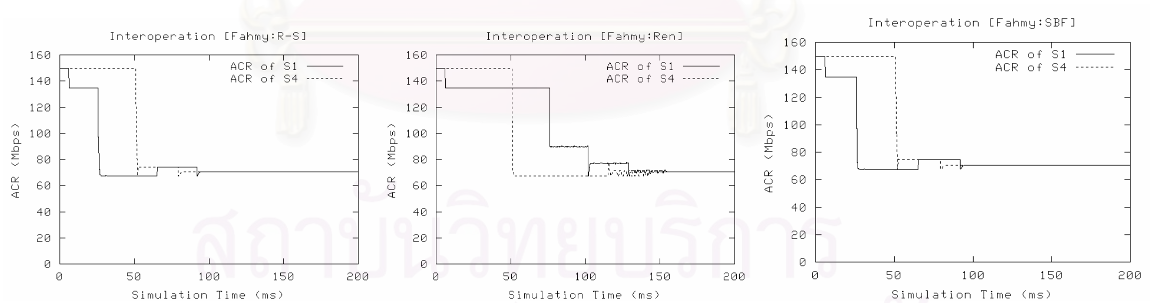
(a) (b) (c)

Figure 6.3 Simulation results of interoperation in WAN configuration using R-S and other consolidation algorithms



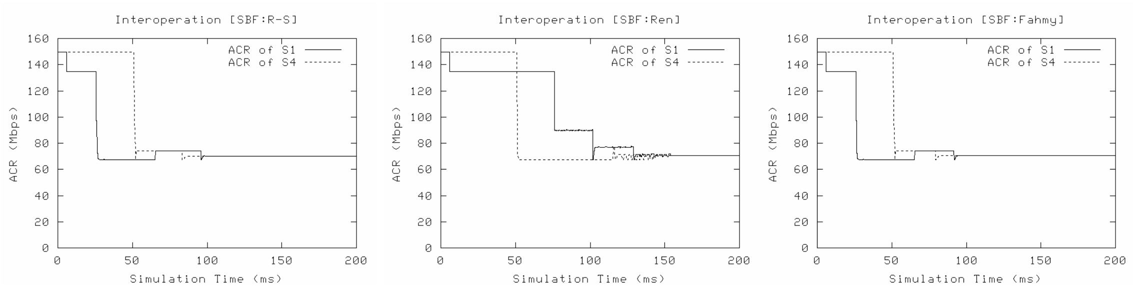
(a) (b) (c)

Figure 6.4 Simulation results of interoperation in WAN configuration using Ren and other consolidation algorithms



(a) (b) (c)

Figure 6.5 Simulation results of interoperation in WAN configuration using Fahmy and other consolidation algorithms



(a) (b) (c)

Figure 6.6 Simulation results of interoperation in WAN configuration using SBF and other consolidation algorithms

6.2.2 LAN/MAN Configuration

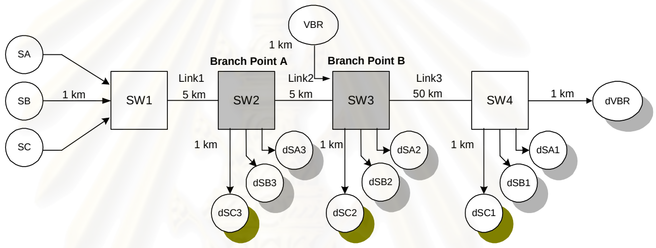


Figure 6.7 Network model II

- Set the switch parameters as in WAN configuration.
- Source SA, SB and SC are persistent sources and their destinations are dSA, dSB and dSC, respectively.
- SA starts sending cells at 0 ms.
- SB and SC starts sending cells at 2.5 ms.
- VBR is a background traffic and its pattern is shown in Figure 6.8.

In LAN/MAN configuration shown in Figure 6.7, link distances were set in a range of tens of kilometers to examine whether the slow response of the wait-for-all algorithm will affect the overall network's response as in WAN case or not. Moreover, we have increased a network complexity by using 3 sessions of multicast (SA, SB, SC and their destinations) and VBR source as a background traffic to make sure that the interoperation can work well in a more complex network. The simulation results are shown in Figure 6.9 to Figure 6.13.

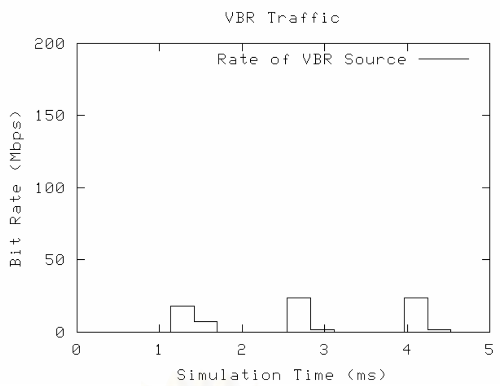


Figure 6.8 VBR traffic pattern

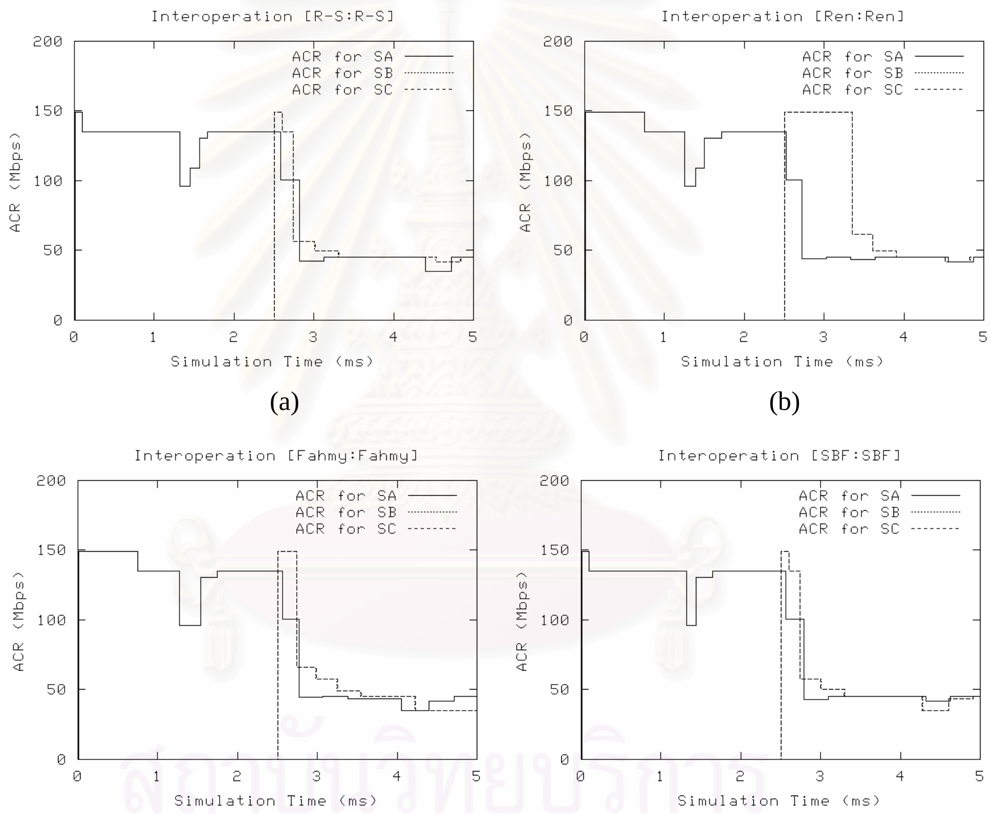
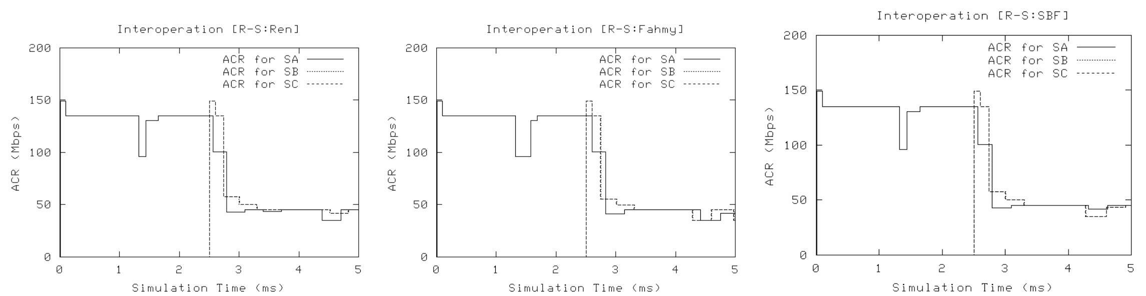
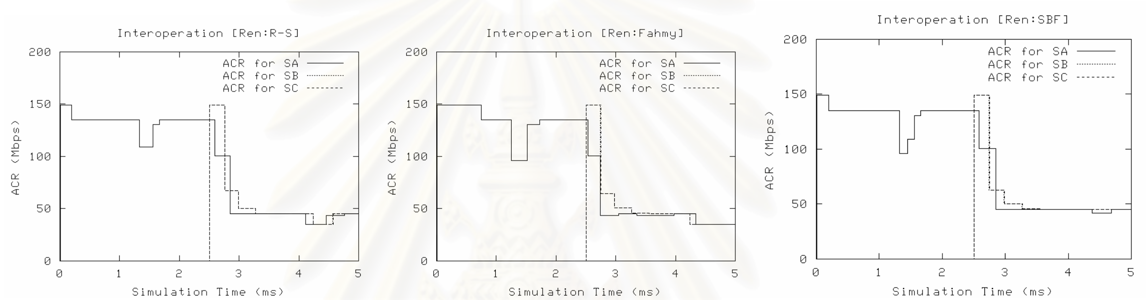


Figure 6.9 Simulation results of LAN configuration using the same consolidation algorithm at branch points



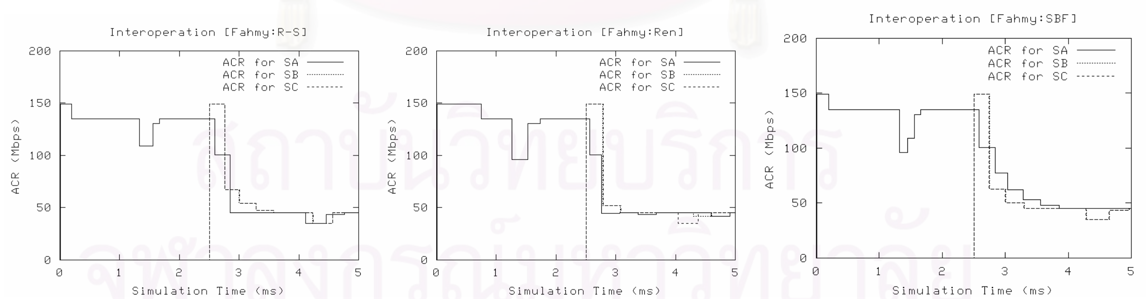
(a) (b) (c)

Figure 6.10 Simulation results of interoperation in LAN configuration using R-S and other consolidation algorithms



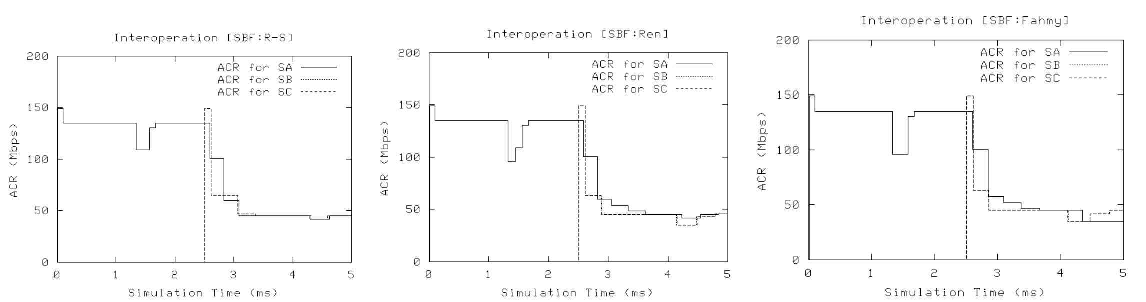
(a) (b) (c)

Figure 6.11 Simulation results of interoperation in LAN configuration using Ren and other consolidation algorithms



(a) (b) (c)

Figure 6.12 Simulation results of interoperation in LAN configuration using Fahmy and other consolidation algorithms



(a) (b) (c)

Figure 6.13 Simulation results of interoperation in LAN configuration using SBF and other consolidation algorithms

The Averaging Interval (AI time) parameter set in rate allocation algorithm (ERICA) has to be taken into account. We set it to be 0.1 ms. Actually, the round trip distance from source to the nearest destinations is 14 km and will take only 0.07 ms. response time for R-S and SBF. From Figure 6.9(a) and Figure 6.9(d), the ACR drops from 150 Mbps to 135 Mbps in 0.1 ms where Fahmy and Ren takes 0.8 ms (The round trip distance from source to the farthest destinations is 144 km and should take only 0.72 ms).

6.3 Discussions

6.3.1 Conventional Network Results

Figure 6.2 shows the results that branch point A and branch point B are implemented with the same consolidation algorithm. We call this kind of network as conventional network. The results of the branch points that are both implemented with R-S are shown in Figure 6.2(a). It is obvious that R-S gives fast response but high consolidation noise The ACR of S1 is reduced promptly from 150 Mbps (ICR) to 135 Mbps (90 % of PCR as preset at the switch) when the first BRM cell from dS3 arrived at S1 (at 6 ms). This response time is in accordance with the equation analyzed in eq.(5.7) which is equal to the round trip of the propagation delay from source, S1, to destination dS3 via branch point A. The ACR value lasts until the first BRM cell from dS2 arrives at S1 (at 26 ms) then the ACR is reduced to 67.5 Mbps (the fair share bandwidth between S1 and S4 at branch point B). After the arrival of the farthest BRM (from dS3) at 76 ms, S1 starts to oscillate and the consolidation noise occurs. This is because the ACR of S1 depends on the BRM cell being sent to it by the branch point A. If S1 receives BRM cell from dS3 at that moment, then it sets ACR to 135 Mbps. On the other hand, if BRM cell is from dS1 or dS2 then ACR of S1 will be set to 67.5 Mbps. Hence, the ACR will oscillate between 67.5 Mbps and 135 Mbps. The ACR of a not wait-for-all analyzed in eq.(5.12) and eq.(5.13) can be applied to describe this circumstance. For a unicast session, the ACR of S4 is dropped to a fair share value after the t_{RTD} of the unicast session is reached (51 ms).

The result of implementing Ren, which is a 'wait-for-all' algorithm, in the branch points is shown in Figure 6.2(b). We see that there is no consolidation noise. However, the ACR of S1 starts to drop from ICR to a fair share rate after BRM cell from dS1 (the farthest) has arrived. It takes 76 ms before starting to approach a fair share rate. This is in line with the response time for wait-for-all analyzed in eq(2). It is equal to a round trip time from source to branch point A plus a propagation time from branch point A to the farthest destination of the lowest branch point level in the multicast session. In addition, equation analyzed in eq.(5.11) can be used to insist the ACR graph in the simulation result. Comparing this slow response to R-S fast response time we can see that Ren will introduce a queue blow up problem, especially in WAN. In the simulation, the time that S1 starts to get to the fair share value is about 50 ms difference between R-S and Ren (26 ms for R-S and 76 ms for Ren). Hence, the queue that will be built up at branch point B is about 4.125 Mbits or 9730 cells (50 ms times 82.5 Mbps (PCR minus fair share rate)) and these cells may be lost if the buffer is insufficiently provided.

For Fahmy, the results are shown in Figure 6.2(c). We see that they exhibit a fast response and no consolidation noise. Notice that during the initial stage, the incoming data rate (150 Mbps) exceeds the capacity of the branch point (135 Mbps). Consequently, there is an overload at branch point A and Fahmy can detect this situation. Hence, it works in a fast response mode and eq.(5.7) can be used to describe this simulation result. The ACR of S1 can be expressed by eq.(5.14) to eq.(5.16).

Similarly for SBF, the response time and ACR shown in Figure 6.2(d) can be expressed by the equations analyzed in eq.(5.7) and eq.(5.17) to eq.(5.18), respectively.

6.3.2 Consolidation Algorithm Interoperation Network Results

Figure 6.3 to Figure 6.6 are the simulation results of the interoperation of the branch points implementing with different consolidation algorithms in WAN configuration. We investigate all possible combination of the algorithms. The notation, e.g. [R-S:Ren], we used on each graph in all figures means branch point A and branch point B are implemented with R-S and Ren, respectively.

Similar to a conventional network, the analyzed equations can also be used to explain the behavior of an interoperation network. However, in order to avoid a repetition in this section we will analyze the results comprehensively as follows.

1. Figure 6.3(a), R-S at branch point A is a fast response algorithm and as explained in the previous section that the response time is dictated by eq.(5.7), hence the ACR of S1 could be

reduced to 135 Mbps very fast (6 ms). However, Ren at branch point B has to wait for the farthest BRM cell before sending a BRM cell back to branch point A. This waiting time can be calculated by using eq.(5.2). Hence, ACR of S1 starts to drop to a fair share rate at 76 ms and oscillates after this time.

2. Figure 6.3(b) is similar to Figure 6.3(a) except for the temporary dropping to a fair share rate of ACR at 26 ms time. This is because Fahmy can detect an overload at branch point B and immediately sending back a BRM cell to branch point A without waiting for BRM from the farthest destination. After that, the next BRM cell received by S1 comes from dS3 so the ACR go abruptly back to 135 Mbps.

3. The interoperation between R-S and SBF in Figure 6.3(c) exhibits a fast response at 26 ms but still exhibits the ACR oscillation.

4. Figure 6.4(a) to Figure 6.4(c) show the same pattern of ACR of S1 regardless of what algorithm is used in branch point B. This is because branch point A has to wait for all branches connecting to it before sending a BRM cell to S1. However, the ACR remains at 150 Mbps until it drops to fair share (do not drop to 135 Mbps before getting to fair share as in Figure 6.3). Note that the dropping point of ACR in Figure 6.4 is faster than that in Figure 6.2(b). The reason is R-S, Fahmy and SBF at branch point B have a fast response so they can send a BRM to branch point A at 26 ms while Ren has to wait for BRM from dS1 which is the farthest destination.

5. The results of using Fahmy and SBF at branch point A are shown in Figure 6.5 and Figure 6.6. Fahmy and SBF overcome R-S and Ren’s drawbacks. The consolidation noise exists in Figure 6.3 and the slow response exists in Figure 6.4 are solved. However, if Ren is used at branch point B, S1 will take longer time than using others before getting to a fair share rate.

For LAN, all combination of algorithms comes out mostly with similar results which there is no consolidation noise for all algorithms used. This is because the slow response algorithm i.e. Ren and Fahmy, they have no noise themselves. While for fast response algorithm i.e. R-S, AI time is longer than the propagation time of all destinations connecting to branch point A, hence R-S virtually work as a wait-for-all algorithm like Ren.
The ACR gets to fair share in a fast time except for Ren and Fahmy (see ACR of SA in Figure 6.11(b) and Figure 6.12(b)). For Fahmy, the response time is as slow as Ren's because there is no overload at branch point A (see Figure 6.9(c) and compare to Figure 6.2(c) in the case that there is an overload at branch point A).

6.3.3 Effects of Asymmetrical RTD

Since the round trip delay plays an important role in determining the effectiveness of the consolidation algorithm and the network performance. Therefore, an asymmetry among round trip delays to leaves from a branch point should be taken into account carefully. An overlook of this parameter may lead to a slow transient response and queue build up at the bottlenecked switch. Consolidation algorithm is a major part in handling this situation. Unfortunately, some algorithms may be impacted by the effect of asymmetrical RTD. For example, although the wait-for-all algorithm (e.g. Ren) may not introduce consolidation noise, but it can not cope with the slow transient response in an asymmetrical RTD network environment. We can see from eq.(5.2) and eq.(5.5) that, the response time of Ren depends on the round trip delay and a number of the branch point level. The response time increases proportionally with the number of branch point in a path. Hence, the response time of the network is determined by the branch or path with the longest delay time.

On the other hand, the not-wait-for-all algorithm (e.g. R-S) does not suffer from the slow response problem since its response time is equal to the shortest round trip delay of the branch point to the destinations. However, the consolidation noise is produced due to loosing a synchronization of RM cells in the network. Hence, if the consolidation algorithms are not well designed, the asymmetrical RTD in the network will affect to their operation and to the network performance.

Consider the impact of an asymmetrical RTD in multicast session to the consolidation algorithm in Figure 6.1, the BRM cells from different downstream branches (dS1, dS2 and dS3) may arrive at branch points (A and B) at different time. This will lead to a consolidation noise and consolidation delay problem. Hence, a consolidation algorithm at the branch point should support not only the synchronization of these BRM cells before sending them to an upstream node but also the minimization of the response time of the network. As described in section 3.4.1 to 3.4.3 of chapter 3 and section 4.2 in chapter 4, R-S, Ren, Fahmy and SBF have difference in characteristics according to RM cell synchronization and response time minimization. The characteristics of each algorithm are summarized in Table 6.1.

Table 6.1 Characteristics of consolidation algorithms

Algorithm	RM cell Synchronization	Response Time Minimization
R-S	No	Yes
Ren	Yes	No
Fahmy	Yes	Yes
SBF	Yes	Yes

From Table 6.1, Since R-S emphasizes on the response time minimization, branch point A forwards the cell from dS1 to the source without waiting BRM cells from dS2 and dS3. Hence, asymmetrical RTD may not affect the response time but rather the BRM cells synchronization which leads to a consolidation noise problem (see Figure 6.2(a) and Figure 6.3(a), (b) and (c)). On the other hand, as Ren concerns more on the BRM-feedback synchronization, it has to wait for BRM cells from dS1, dS2 and dS3 for synchronizing all branches. Therefore, the response time in asymmetric round trip delay session is governed by a branch with the longest delay time (see Figure 6.2(b) and Figure 6.4(a), (b) and (c)). For Fahmy, the algorithm operates like Ren in case that there is no overload situation at the branch point. It means that Fahmy is affected by asymmetrical RTD. But if the overload is detected, the branch point will forward the BRM cell upstream immediately to reduce the source rate. Therefore, Fahmy is not so much sensitive as Ren in terms of response time to the asymmetrical RTD. Figure 6.6 exhibits the same result as in Figure 6.5 (Fahmy with overload situation at branch point). Hence, we can say that SBF is robust to network environment and insensitive to the asymmetrical RTD in multicast session. So far, the simulation results of all cases of the consolidation algorithm interoperation are summarized in Table 6.2 and Table 6.3.

Table 6.2 Simulation results for the network model I (WAN) and model II (LAN) using the same consolidation algorithm at branch point A and branch point B

Branch point A	Branch Point B	Transient Response Time* (ms)		Consolidation Noise		Time to get to Fair Share rate (ms)		Impact of Asymmetrical Round Trip Delay	
		WAN	LAN	WAN	LAN	WAN**	LAN***	WAN	LAN
R-S	R-S	6	0.1	Yes	No	-	0.1	Yes	No$
Ren	Ren	76	0.8	No	No	> 76	0.8	Yes	No$
Fahmy	Fahmy	6	0.8	No	No	26	0.1	No$	No$
SBF	SBF	6	0.1	No	No	26	0.1	No	No

** The earliest time that the network feedbacks information to adjust the source rate.*

*** The time that S1 and S4 get to the fair share value.*

**** The time that all sources get to the fair share value after the appearance of SB and SC at 2.5 ms.*

$ Actually these algorithms are affected by asymmetrical round trip delay but insignificantly

Table 6.3 Simulation results for the network model I (WAN) and model II (LAN) using different consolidation algorithms at branch point A and branch point B

Branch point A	Branch Point B	Transient Response Time*(ms)		Consolidation Noise		Time to get to Fair Share rate (ms)		Impact of Asymmetrical Round Trip Delay	
		WAN	LAN	WAN	LAN	WAN	LAN***	WAN	LAN**
R-S	Ren	6	0.1	Yes	No	-	0.1	Yes	No[$]
	Fahmy	6	0.1	Yes	No	-	0.1	Yes	No[$]
	SBF	6	0.1	Yes	No	26[**]	0.1	Yes	No[$]
Ren	R-S	26	0.2	No	No	26	0.2	Yes	No[$]
	Fahmy	26	0.8	No	No	26	0.2	Yes	No[$]
	SBF	26	0.2	No	No	26	0.2	Yes	No[$]
Fahmy	R-S	6	0.2	No	No	26	0.1	No[$]	No[$]
	Ren	6	0.8	No	No	> 76	0.1	No[$]	No[$]
	SBF	6	0.2	No	No	26	0.1	No[$]	No[$]
SBF	R-S	6	0.1	No	No	26	0.1	No	No
	Ren	6	0.1	No	No	> 76	0.1	No	No
	Fahmy	6	0.1	No	No	26	0.1	No	No

** The earliest time that the network feedbacks information to adjust the source rate.*

*** Though consolidation noise exists, but S1 gets to fair share rate during 26 - 76 ms period.*

**** The time that all sources get to the fair share value after the appearance of SB and SC at 2.5 ms.*

 # There is no significant difference for each algorithm because of the short link distance in LAN configuration.

$ Actually these algorithms are affected by asymmetrical round trip delay but insignificantly.

We can conclude from Table 6.2 and Table 6.3. that the most upper stream branch point (branch point A) is the most important one in the interoperation of consolidation algorithm in point-to-multipoint network, especially in WAN configuration. It plays a critical role in determining the network performance. The response time and consolidation noise of the interoperated network tend to follow the characteristics of consolidation algorithm used at the most upper stream branch point. We recommend to implement a fast response and low consolidation noise algorithm such as Fahmy or SBF at this branch point in order to avoid the consolidation noise and get a faster response. We found that the characteristics of the consolidation algorithm implemented at the lower stream branch point (branch point B) affects insignificantly to the network performance. Hence, a consolidation algorithm, which is simple and easy to implement, should be used at this branch point.

For the effect of asymmetrical RTD in interoperated network, for WAN, R-S and Ren algorithms are sensitive to this parameter while Fahmy is affected only in the case that the overload does not exist at the branch point. We can say that Fahmy is partly sensitive. SBF operates independently with this parameter. Similar to what have been analyzed in the case of response time and ACR, an asymmetrical RTD may significantly affect the network performance if the consolidation that is sensitive to this parameter is used at the most upper stream branch point. Otherwise, the effect is not obvious. In LAN/MAN, the impact of an asymmetrical round trip delay is insignificant because the difference in time delay is not much. Therefore, we can conclude that in LAN/MAN environment, the performance of the consolidation algorithm-interoperated network is affected insignificantly regardless of which consolidation algorithms used at any branch point.

CHAPTER 7

Conclusion and Future Work

We have developed congestion and flow control algorithms for ABR multicast connections that focus on response time, consolidation noise, fairness and link utilization of the developed schemes. Two consolidation algorithms were proposed, namely, RQB and SBF. For RQB, it has been designed to compromise between the response time and accuracy. The algorithm can track the changing traffic in the network and utilize the available network bandwidth efficiently. It is also capable to adapt itself to work in fast response or high accuracy mode according to the network condition. The simulation results show that it works, in some aspects, better than the previous proposed algorithms. RQB algorithm works well in the sense of meeting the major aspect required in the point-to-multipoint ABR connections i.e. fast response and low consolidation noise. However, it introduces a little more complexity to the algorithm by using the QL (Queue Length) field in RM cell. Actually, ATM Forum has defined QL in RM cell but its value is set to zero. Therefore, there are still some rooms for improvement of this algorithm in terms of the implementation complexity and avoiding the use of QL field. Hence, SBF was proposed for this reason. The main functional operation of the algorithm is that it keeps track of the most congested branch. For this, the branch point does not wait for all BRM cells but selectively sends BRM cell, which contains the least ER value to the source. We have extensively performed an evaluation (by simulation) through various configurations and traffic patterns. Simulation results show that SBF algorithm meets at most consideration issues for Point-to-Multipoint ABR Service comparing to algorithms that have ever been proposed. Besides fast response, low consolidation noise and low complexity, the BRM/FRM ration is controlled to be 1 or nearly. It can take care of overload condition in downstream branches and can utilize, during the transient state, the available bandwidth, especially left from the VBR source in the network. It is also insensitive to the number of branches and branch point level in the network and complies with the ATM Forum guidelines. Moreover, mathematical analysis for many types of consolidation algorithm has been investigated. The equations for approximating response time and allowed cell rate of the source in various network topologies have been analyzed.

The analyzed equations were proved to exhibit the results in line with the result from simulation. We can use these equations to describe the behavior of the consolidation algorithms in the multicast ABR connections in terms of response time, consolidation noise and effect of asymmetrical round trip delay.

The interoperation issue has been extensively investigated. Four representative consolidation algorithms have been selected to interoperate in both LAN/MAN and WAN network configurations. We found that in WAN, the most upper stream branch point (branch point A) plays an important role in determining the performance of the interoperated network. The overall response time and consolidation noise of the interoperated network is likely to be the same as the characteristic of consolidation algorithm used at this branch point. For the effect of asymmetrical RTD aspect, R-S and Ren are sensitive to this parameter while Fahmy is partly sensitive. SBF is independent on this parameter. Similarly to response time and ACR case, an asymmetrical RTD may significantly affects to the network performance if the consolidation algorithm, which is sensitive to this parameter, is used at the most upper stream branch point. Otherwise, the effect is not obvious. Hence, the most upper stream branch point should be implemented with a fast response and low noise algorithm such as Fahmy or SBF. The network performance is not much affected by the characteristic of the consolidation algorithms implemented at the lower stream branch points. Therefore, a simple and easy to implement consolidation algorithm is recommended at these branch points. In LAN/MAN, there is no significant difference in both response time and consolidation noise regardless of which consolidation algorithms were interoperated. The impact of an asymmetrical RTD is also insignificant because the difference in time delay of branches in the network is not much. We can say that the performance of the consolidation algorithm-interoperated network is not much affected by the consolidation algorithms used at any branch points. Therefore, in LAN/MAN environment the implementation complexity of the consolidation algorithm should be the major issue to be considered for the interoperation of multicast ABR.

Future Work

There are some important issues relating to the proposed consolidation algorithms that should be further investigated. The robustness and scalability of the proposed algorithm needs to be further tested with more number of source and destination end systems. Because we used the data traffic in simulation, it would be interesting if a more realistic input traffic patterns such as video or self-similar streams are used. Analytical model of the consolidation algorithms in multicast ABR is another substantial area. Although we have provided an initial work for the allowed cell rate and response time calculation, some other network performance parameters such as the buffer size needs to be analyzed. In addition, ABR service parameters (ICR, MCR, RIF, TBE etc.) should be set to a wide range of values. This will make the model more realistic and can be used to predict the network performance or used for sizing the network elements.

REFERENCES

[1] A. Charny, D. Clark and R. Jain, "Congestion control with explicit rate indication," AF-TM 940692, July 1994. Updated version published in Proc. ICC'95, June 1995.

[2] A. Charny, K. Ramakrishnan, "Time-Scale Analysis of Explicit Rate Allocation in ATM Networks," Proceedings of the IEEE Infocom 1996, San Francisco, CA, March 1996, Vol. 3, pp.1182-1189.

[3] A. Charny, K. Ramakrishnan, A. Lauck, "Scalability Issues for Distributed Explicit Rate Allocation in ATM Networks," Proceedings of the IEEE Infocom 1996, San Francisco, CA, March 1996, Vol. 3, pp. 1198-1205.

[4] A. Charny, K. Ramakrishnan, A. Lauck, "Time-Scale Analysis and Scalability Issues for Explicit Rate Allocation in ATM Networks," IEEE/ACM Transactions on Networking, Vol. 4, No. 4, August 1996, pp. 569-581.

[5] Abhay K. Parekh and Robert G. Gallager, "A Generalized Processor Sharing approach to flow control in integrated services networks - the multiple node case," Proc. IEEE INFOCOM'93, pp. 521-530, San Francisco, CA, March 1993

[6] Alan Demers, Srinivasan Keshav and Scott Shenker, "Analysis and simulation of fair queuing algorithm," Journal of Internet Working Research and Experience, pp. 3-26, October 1990. Also in Proc. ACM SIGCOMM'89, pp. 3-12.

[7] A. Arulambalam, X. Chen, N. Ansari, "Allocating Fair Rates for Available Bit Rate Service in ATM Networks," IEEE Communications Magazine, Vol. 34, No. 111 November 1996, pp. 92-100.

[8] ATM Forum, "Traffic Management Specification Version 4.0," AF-TM-0056.000, April 1996.

[9] B.G. Kim and P. Wang, "ATM Networks: Goals and Challenges," ACM Communications, Feb. 1995, Vol 2, No. 2.

[10] D. Bertsekas, R. Gallager, "Data Networks," Second Edition, Prentice Hall Inc.,New Jersey, 1992.

[11] F. Bonomi, K. Fendick, "The Rate-Based Flow Control Framework for the Available Bit Rate ATM Service," IEEE Network Magazine, Vol. 9, No. 2, March/April 1995, pp. 25-39.

[12] F. Bonomi, D. Mitra, J. Seery, "Adaptive Algorithm for Feedback-Based Flow Control in High Speed, Wide-Area ATM Network," IEEE Journal on Selected Areas in Communications, Vol. 13, No. 7, September 1995,pp. 1267-83.

[13] G. Kesidis, "ATM Network Performance," Kluwer Academic Press, Boston, MA, 1996.

[14] G.L. Chaudhury, D.M. Lucantoni, and W. Whitt, “On the effectiveness of effective bandwidths for admission control in ATM networks,” ITC 14, pp. 411-420, June 1994.

[15] H. Ohsaki, M. Murata, H. Suzuki, C. Ikeda, H. Miyahara, “Rate-Based Congestion Control for ATM Networks,” Computer Communication Review, Vol. 25, No. 2, April 1995, pp. 60-72.

[16] H. Ohsaki, M. Murata, H. Suzuki, C. Ikeda, H. Miyahara, “Performance Evaluation of Rate-Based Congestion Control Algorithms in Multimedia ATM Networks,” Proceeding of IEEE Globecom 1995, Singapore, November 1995, Vol. 2, pp. 1243-1248.

[17] H. Ohsaki, M. Murata, H. Suzuki, C. Ikeda, H. Miyahara, “Analysis of Rate-Based Congestion Control Algorithms for ATM Networks Part 1: Steady-State Analysis,” Proceedings of IEEE Globecom 1995, Singapore, November 1995, Vol. 1, pp. 296-303.

[18] H. Ohsaki, M. Murata, H. Suzuki, C. Ikeda, H. Miyahara, “Analysis of Rate-Based Congestion Control Algorithms for ATM Networks Part 2: Initial Transient State Analysis,” Proceedings of IEEE Globecom 1995, Singapore, November 1995, Vol. 2, pp. 296-303.

[19] H. Saito et al, “Performance Issues in Public ABR Service," IEEE Communtcations Magazine,” Vol. 34, No. 11, November 1996, pp. 40-48.

[20] H. T. Kung Harvard University, “The FCVC (Flow Controlled Virtual Channels) Proposal for ATM Networks,” Proc. 1993 International conference on Network Protocols, San Francisco, California, Oct 19-22,1993, pp. 116-127, Modified April 8,1994.

[21] H. T. Kung et al, “Flow Controlled Virtual Connections Proposal for ATM Traffic Management,” AF-TM 94-0632R2, September 1994.

[22] H. Y. Tzeng and K. Y. Siu, “On Max-min fair congestion control for multicast ABR service in ATM,” IEEE Journal on Selected Areas in Communications, vol. 15, no. 3, Apr. 1997.

[23] ITU-T, “Broadband Reference Model and Its Application,” Recommendation I.321, Geneva, 1991.

[24] Kai-Yeung Siu and Hong-Yi Tzeng, “Congestion control for multicast service in ATM networks,” Proceedings of the IEEE GLOBECOM, vol.1. pp. 310-314, 1995.

[25] K. K. Ramakrishnan and R. Jain, “A binary feedback scheme for congestion avoidance in computer networks,” ACM Trans. Comput. Sys., VoL 8, No. 2, pp. 158-181, 1990.

[26] K. K. Ramakrishnan, Peter Newman, “Integration of Rate and Credit Schemes for ATM Flow Control,” IEEE Network March/April 1995.

[27] K. Siu, H. Tzeng, "Intelligent Congestion Control for ABR Service in ATM Networks," Computer Communication Review, Vol. 24, No. 5, October 1995, pp. 81-106.

[28] L. Kalampoukas, A. Varma, K.K. Ramakrishnan, "An Efficient Rate Allocation Algorithm for ATM Networks Providing Max-Min Fairness," Proceedings of the 6th IFIP International Conference on High Performance Networking, HPN'95, September 1995.

[29] Lawrence Roberts, "Enhance PRCA (Proportional Rate Control Algorithm)," AF-TM 94-0916, September 1994.

[30] Lawrence Roberts, "Rate based algorithm for point to multipoint ABR service," ATM Forum Contribution 94-0772R1, November 1994.

[31] M Baumann, "YATS – Yet Another Tiny Simulator User's Manual for Version 0.3," http://www.ifn.et.tu-dresden.de/TK/yats/yats.html, Communication Laboratory, Dresden University of Technology, 1996.

[32] M. Ritter, "Network Buffering Requirements of the Rate-Based Control Mechanism for ABR Service," Proceedings of the IEEE Infocom 1996, San Francisco, CA, March 1996, Vol. 3, pp. 1190-1197.

[33] M. Hluchyj et al, "Closed-Loop Rate-Based Traffic Management," AF-TM 94-021]R3, April 1994.

[34] M. Hluchyj and N. Yin, "On closed-loop rate control for ATM networks," Proc. INFOCOM 94, pp. 99-108,1994.

[35] M. Tufail and B. Cousin. 1996. "Timer imposed and priority supported (TIPS) congestion control scheme for point-to-multipoint connections in ATM." Proc. of 11th Congres De Nouvelles Architectures pour les Communications (DNAC) Large Bande: Internet ou RNIS, France, 65-79.

[36] Michael Ritter, "The Effect of Bottleneck Service Rate Variations on the Performance of the ABR Flow Control," INFOCOM '97, Proceedings IEEE Volume: 2, 1997, Page(s): 815 -822 vol.2.

[37] N. Ghani, J. W. Mark, "Dynamic Rate-Based Control Algorithm for ABR Service in ATM Networks," Proceedings of IEEE Globecom 1996, November 1996, London, UK, Vol. 2, pp. 1074-1079.

[38] N. Ghani, J. W. Mark, "Queueing Analysis of a Distributed Explicit Rate Allocation Algorithm for ABR Services," Proceedings of the IEEE Infocom 1997, Kobe, Japan, April 1997.

[39] N. Yin, M. Hluchyj, "A Dynamic Rate Control Mechanism for Source Coded Traffic in a Fast Packet Network," IEEE Journal on Selected Areas in Communications, Vol. 9, No. 7, September 1991, pp. 1005-1012.

[40] Nada Golmie, Alfred Koenig, David Su, “The NIST ATM Network Simulator,” Computer Science Laboratory, Gaithersburg, MD 20899, Aug 1995.

[41] Nanying Yin, “Analysis of a Rate-based Traffic Management Mechanism for ABR Service,” GLOBECOM '95, Global Telecommunications Conference, 1995. IEEE Volume: 2, 1995, Page(s): 1076 -1082 vol.2.

[42] N. Rangsinoppamas, T. Jaruvitayakovit, W. Sa-Niamsak, and P. Prapinmongkolkarn, “Compromised Consolidation Algorithm for Point-to-Multipoint ABR Service in ATM Networks,” Proceedings of the Applied Telecommunications Symposium (ATS'99), San Diego,1999.

[43] N. Rangsinoppamas T. Jaruvitayakovit, W. Sa-Niamsak, P.Pavarangkul and P. Prapinmongkolkarn, “Consolidation Algorithm Interoperability in Point-to-Multipoint ABR Services in ATM Networks,” Proceeding of the IFIP-TC6/European Commission International Conference (Networking 2000), Paris, France, May 2000.

[44] N. Rangsinoppamas, T. Jaruvitayakovit and P. Prapinmongkolkarn, “Performance Analysis and Interoperability Issue of Consolidation Algorithms in Point-to-Multipoint ABR Services,” to be published in An International Journal of Communication Systems, Wiley-Interscience Publication.

[45] N. Rangsinoppamas, T. Jaruvitayakovit and P. Prapinmongkolkarn, “Interoperation and Analysis of Consolidation Algorithm for Point-to-Multipoint ABR Service in ATM Networks” conditional acceptance for publication in IEICE Transactions on Communications, Japan.

[46] Omar Ait-Hellal, Eitan Altman, “Performance Evaluation of Congestion Phenomena in the Rate Based Flow Control mechanism for ABR,” INFOCOM '99, Proceedings. IEEE Volume: 2, 1999, Page(s): 819 -826 vol.2.

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

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

[49] 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, pp. 1723-1738.

[50] R. Jain, S. Kalyanaraman, S. Fahmy, R. Goyal, “Source Behaviour for ATM ABR Traffic Management: An Explanation,” IEEE Communications Magazine, Vol. 34, No. 11, November 1996, pp. 50-57.

[51] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and R. Viswanathan, “ERICA switch algorithm: A complete description,” ATM Forum 96-1172, Aug. 1996.

[52] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, F. Lu, "ERICA+: Extensions to the ERICA Switch Algorithm," ATM Forum- Traffic Management 95-1346, October 1995.

[53] R. Jain, S. Fahmy, S. Kalyanaraman, R. Goyal, "ABR Switch Algorithm Testing: A Case Study with ERICA," ATM Forum-Traffic Management 96-1267, October 1996.

[54] S. Mascolo, D. Cavendish, M. Gerla, "ATM Rate Based Congestion Control Using a Smith Predictor: An EPRCA Implementation," Proceedings of the IEEE Globecom 1996, London, UK, November 1996, pp. 569-576.

[55] S. Shenker, "Making greed work in networks: A game-theoretical analysis of switch service disciplines," Proc. ACM SIGCOMM'94, pp. 47-57, London, UK, August 1994.

[56] Shivkumar Kalyanaraman, Raj Jain, Rohit Goyal, Sonia Fahmy and Seong-cheol Kim, "Use-it or Lose-it policies for the Available Bit Rate (ABR) service in ATM networks," Journal of Computer Networks and ISDN system, pp. 25, January 1997.

[57] Sonia Fahmy, Raj Jain, Sivkumar Kalyanaraman, Rohit Goyal, Bobby Vandalore, Sastri Kota and Pradeep Samudra, "Feedback consolidation algorithm for ABR point-to-multipoint connections," ATM Forum Contributions 97-0615, July 1997 and IEEE INFOCOM'98 March 1998.

[58] Sonia Fahmy, "Traffic Management for Point-to-Point and Multipoint Available Bit Rate (ABR) Service in Asynchronous Transfer Mode (ATM) Networks," Dissertation, The Ohio-State University, 1999

[59] Tianji Jiang, Ellen W. Zegura and Mostafa Ammar. 1998. "Improved consolidation algorithms for point-to-multipoint ABR service." 0-7803-4874-5/98, IEEE.

[60] W. Ren, K-Y Siu, and H.Suzuki, "On the performance of congestion control algorithms for multicast ABR service in ATM," In Proceedings of IEEE ATM'96 Workshop, San Francisco, Aug. 1996.

[61] URL for the simulator used in this dissertation. http://ee.kmitnb.ac.th/~nrr/Simulator/simulator.html

APPENDICES

Appendix 1

Simulation Program

There are many free and open-source simulation programs available online e.g. [31], [40]. In this dissertation, because of its tiny size, we have developed the simulation tool by using of YATS (Yet Another Tiny Simulator) software from Dresden University of Technology, Germany [31], as a core program. The detail of the developed version is available at http://ee.kmitnb.ac.th/~nrr/Simulator/simulator.html [61] and the original YATS version is at http://www.ifn.et.tu-dresden.de/TK/yats/yats.html.

A1.1 Main Program

The main program used in this thesis is modified from YATS program. A flowchart of the main program is shown in Figure A1.1

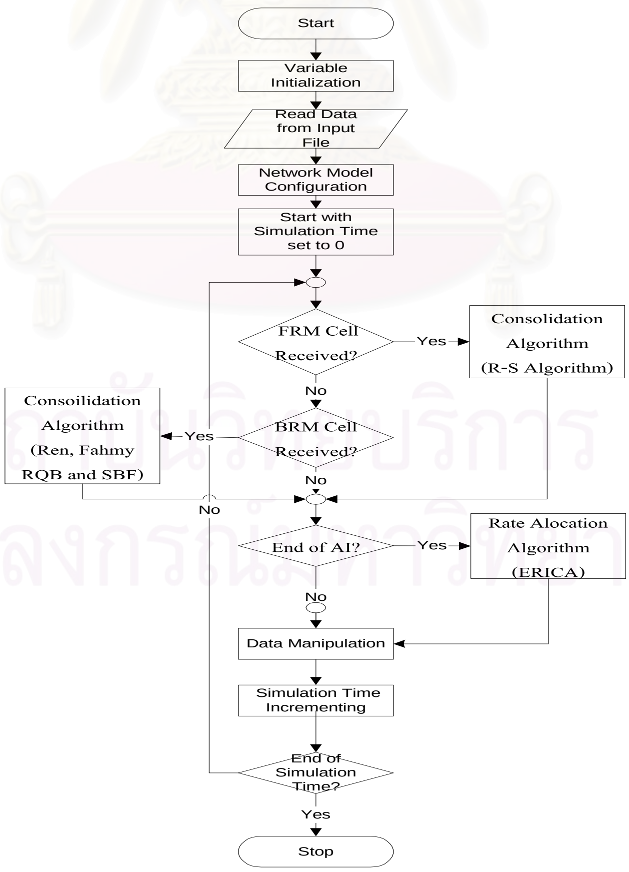


Figure A1.1 Main program flowchart

A1.2 An Example of Input File

Chain Configuration Algorithm SBF

```
// 1: Multicast 1 sources => 3 destinations [Parking lot]
// 2: Unicast 1 source  => 1 destination  [Point-to-Point]
 MaxLinkCR = 149.76;

vari = 1;
varAlgor = SBF, I=88;
varAvg_Interval = 100;

{   // data source
    GmdpStop gmdp[1]: NSTAT=2, DELTA=(1,1), EX=(10,50),
        TRANS=(0,1,1,0), VCI=1, OUT=abrsrc[1];

    // MULTICAST ABR source
    AbrSrc abrsrc[1]: BUFF=5, BSTART=2, RIF=1.0, /*RDF=1.0,*/
        LINKCR=149.76, MCR=0, PCR=149.76, ICR=149.76, FRTT=1,
    MP_ROUTE=(3,
            (5, abrsw1, abrsw2 BP, abrsw3 BP, abrsw4, abrdest[i]),
            (4, abrsw1, abrsw2,    abrsw3 BP, abrdest[i+1]),
            (3, abrsw1, abrsw2 BP, abrdest[i+2])),
    AUTOCONN, OUTCTRL=gmdp[1]->Start, OUTDATA=linefw1[1];

    // delay forward (line)
    Leitung linefw1[1]: DELAY=I, OUT=abrsw1->InpDATA[1];
    Leitung linefw2[1]: DELAY=I, OUT=abrsw2->InpDATA[1];

    // delay backward (line)
    Leitung linebw1[1]: DELAY=I, OUT=abrsrc[1]->BRMC;
    Leitung linebw2[1]: DELAY=I, OUT=abrsw1->InpBRMC[1];

    // ABR sink
    AbrSink abrdest[1]: AI=Avg_Interval, LINKCR=149.76, TARGUTIL=0.9,
        OUTBRMC=linebw5[1], OUTDATA=datasink[1];
    AbrSink abrdest[2]: AI=Avg_Interval, LINKCR=149.76, TARGUTIL=0.9,
```

```
        OUTBRMC=linebw4[2], OUTDATA=datasink[2];

    Senke datasink[i];
    Senke datasink[i+1];
}

// *** ABR MUX
{
    AbrSwitch abrsw1: NINP=1, MAXVCI=5, BUFFCBR=100, BUFFABR=300000,
            AI=Avg_Interval, NOUTP=1, LINKCR=(149.76),
            ERICA, MMFairness,
            TARGUTIL=0.9,
            Algorithm = Algor,

    AbrSwitch abrsw2: NINP=1, MAXVCI=5, BUFFCBR=100, BUFFABR=300000,
            AI=Avg_Interval, NOUTP=2, LINKCR=(149.76,149.76),
            ERICA, MMFairness,
            TARGUTIL=0.9,
            Algorithm = Algor,

    AbrSwitch abrsw3: NINP=2, MAXVCI=5, BUFFCBR=100, BUFFABR=300000,
            AI=Avg_Interval, NOUTP=2, LINKCR=(149.76,149.76),
            ERICA, //MMFairness,
            TARGUTIL=0.9,
            Algorithm = Algor,

    AbrSwitch abrsw4: NINP=1, MAXVCI=5, BUFFCBR=100, BUFFABR=300000,
            AI=Avg_Interval, NOUTP=2, LINKCR=(149.76,149.76),
            ERICA, MMFairness,
            TARGUTIL=0.9,
            Algorithm = Algor,
}
Sim->Run SLOTS=106000, DOTS = 1000;
```

A1.3 RQB Algorithm

/* Rate Queue Balance Algorithm V.5 (Final Version)*/

```
aVci = MP_BackwardPointer[aVci];

if (BRMReceived[aVci][j] < 1)
   {
    BRMReceived[aVci][j] = 1;
    ++NumOfBRMsRec[aVci];
   }

/* Increment M counter */
COUNTER[aVci][j]++;
MPCI[aVci][j] = pc->CI;
MPNI[aVci][j] = pc->NI;
MPER[aVci][j] = pc->ER;
MPQL[aVci][j] = pc->QL;

/* MER = min(MPER for all branch, comp_ERICA) */
/* MQL = max(MPQL for all branch, QL of this switch) */
ptr = abr_table[aVci];
MER[aVci] = ptr->forwER;
MCI[aVci] = MNI[aVci] = 0;
MQL[aVci] = 0;
for(i=1; i<=Number_Of_Branches[aVci]; i++)
   {
    if (MER[aVci] > MPER[aVci][i])
       MER[aVci] = MPER[aVci][i];
    MCI[aVci] = MCI[aVci] || MPCI[aVci][i];
    MNI[aVci] = MNI[aVci] || MPNI[aVci][i];
    if (MQL[aVci] < MPQL[aVci][j])
       MQL[aVci] = MPQL[aVci][j];
   }

minER = compERICA_for_all_branches(pc);
if (MER[aVci] > minER)
   MER[aVci] = minER;
if (abr_queue > MQL[aVci])
   MQL[aVci] = abr_queue;

if (NumOfBRMsRec[aVci] == Number_Of_Branches[aVci])
   {
    /* New MER_exect */
```

```
            pc->ER = LastER[aVci] = MER[aVci];
            pc->CI = MCI[aVci];
            pc->NI = MNI[aVci];
            pc->QL = LAST_QL[aVci] = MQL[aVci];
            pc->vci = aVci;
            compRelative_for_all_branches(pc);
            *inpBRM_ptr++ = pc;      /** store BRM cell **/
            --FRMminusBRM[aVci];

            NumOfBRMsRec[aVci] = 0;
            for (i=0; i<max_vci ;i++)
               BRMReceived[aVci][i] = 0;

           }

         else
           {
            if (LastER[aVci] > MER[aVci] && LAST_QL[aVci] < MQL[aVci])
               RAI = 0.75;
            else if (LastER[aVci] < MER[aVci] && LAST_QL[aVci] > MQL[aVci])
               RAI = 0.25;
            else
               RAI = 0.5;

            RandomValue = ((double)(my_rand() % 1000) / 1000);
            if (RandomValue < RAI && FRMminusBRM[aVci] > 0)
               {
                pc->ER = LastER[aVci] = MER[aVci];
                pc->CI = MCI[aVci];
                pc->NI = MNI[aVci];
                pc->QL = LAST_QL[aVci] = MQL[aVci];
                pc->vci = aVci;
                compRelative_for_all_branches(pc);
                --FRMminusBRM[aVci];
                *inpBRM_ptr++ = pc;      /** store BRM cell **/
               }
            else
               delete pc;
           }
        break;

      default:      break;
      }
      return;
}
```

A1.4 SBF Algorithm

/* Selective BRM Feedback */

```
j=0;
while (branch_table[j]->output_vci != aVci)
      ++j;
j = branch_table[j]->branch_No;

aVci = MP_BackwardPointer[aVci];

if (pc->ER < MER[aVci] || Num_of_Branch_Rec[aVci] == j)
  {

      if (pc->ER == MER[aVci] && FRMminusBRM[aVci] <=0) {
        delete pc;
      }
      else {
        MER[aVci] = pc->ER;
        Num_of_Branch_Rec[aVci] = j;

        pc->vci = aVci;
        minER = compERICA_for_all_branches(pc);
        if (pc->ER > minER)
            pc->ER = minER;
        compRelative_for_all_branches(pc);

        *inpBRM_ptr++ = pc;      // store BRM cell
        --FRMminusBRM[aVci];

      }
  }
else  delete pc;

break;
```

Appendix 2

ABR Flow Control As Per TM Specification Version 4.0 [8]

A2.1 Introduction

In the ABR service, the source adapts its rate to changing network conditions. Information about the state of the network like bandwidth availability, state of congestion, and impending congestion, is conveyed to the source through special control cells called Resource Management Cells (RM-cells). The following sections specify the format and contents of the RM-cell, the source, destination, and switch behavior, and the parameters used in the service. Optional segmentation of networks, support for virtual paths, and a framework for point-to-multipoint behavior is also specified.

A2.2 ABR Service Parameters

This section defines the parameters which are used to implement ABR flow-control on a per-connection basis. All parameters are defined, including those which are actually constants and not altered by signaling.

A2.2.1 Parameter Descriptions

Table A2.1 ABR parameter descriptions

Label	Description	Units and range
PCR	The Peak Cell Rate, PCR, is the cell rate which the source may never exceed.	In Cells/Sec, See Note 1 for range
MCR	The Minimum Cell Rate, MCR, is the rate at which the source is always allowed to send.	In Cells/Sec, See Note 1 for range
ICR	The Initial Cell Rate, ICR, is the rate at which a source should send initially and after an idle period.	In Cells/Sec, See Note 1 for range
RIF	Rate Increase Factor, RIF, controls the amount by which the cell transmission rate may increase upon receipt of an RM-cell.	RIF is a power of two, ranging from 1132768 to 1.
Nrm	Nrm is the maximum number of cells a source may send for each forward RM-cell.	Power of 2 Range: 2 to 256
Mrm	Mrm controls allocation of bandwidth between forward RM-cells, backward RM-cells, and data cells.	Constant fixed at 2
RDF	The Rate Decrease Factor, RDF, controls the decrease in the cell	RDF is a power of 2 from

	transmission rate.	1132,768 to 1
ACR	The Allowed Cell Rate, ACR, is the current rate at which a source is allowed to send.	Units: Cells/Sec
CRM	Missing RM-cell count. CRM limits the number of forward RM-cells which may be sent in the absence of received backward RM-cells.	CRM is an integer. Its size is implementation specific.
ADTF	The ACR Decrease Time Factor is the time permitted between sending RM-cells before the rate is decreased to ICR.	Units: seconds ADTF range: .01 to 10.23 sec: with granularity of 10 ms.
Trm	Trm provides an upper bound on the time between forward RM-cells for an active source.	Units: milliseconds Trm is 100 times a power of two Range: $100*2^{-7}$ to $100*2^{0}$
FRTT	The Fixed Round-Trip Time, FRIT, is the sum of the fixed and propagation delays from the source to a destination and back.	Units: 1 microseconds Range: 0 to 16.7 seconds
TBE	Transient Buffer Exposure, TBE, is the negotiated number of cells that the network would like to limit the source to sending during startup periods, before the first RM-cell returns.	Units: Cells Range: 0 to 16,777,215
CDF	The Cutoff Decrease Factor, CDF, controls the decrease in ACR associated with CRM.	CDF is zero, or a power of two in the range 1/64 to 1
TCR	The Tagged Cell Rate, TCR, limits the rate at which a source may send out-of-rate forward RM-cells.	TCR is a constant fixed at 10 cells/second

Note 1: Rates are signaled as 24 bit integers which have a minimum value of zero, and a maximum value of 16,777,215. However, RM-cells use a 16-bit floating point format (see Section A2.3.2) which has a maximum value of 4,290,772,992.

A2.2.2 Signaled Parameters

The following parameters are to be signaled and negotiated separately during connection establishment. If any parameter but PCR is unspecified by the source, the first switch will fill in the default value (before negotiation). MCR is optionally negotiable; if MCRmin is missing, then MCR is not negotiable.

Table A2.2 Mandatory parameters to be signaled

Name	Negotiation	Default
PCR	down	mandatory
MCR	down to MCRrnin if MCRmin is signaled, else no	0
ICR	down	PCR
TBE	down	16,777,215
FRTT	accumulated	Note 1
RDF	down	1/16
RIF	down	1/16

Note 1: FRTT (Fixed Round-Trip Time) should be set by the source to the fixed source delay. FRTT is then accumulated during the call setup. FRTT is used to determine other parameters (see Section A2.2.4). It should be the sum of all the RM-cell fixed delays plus propagation delays in the round trip call path.

Note 2: Because of the downward negotiation of RIF and RDF, a given switch may not be able to support the RIF and RDF values selected by switches farther from the source. This may occur because the RIF or RDF value is smaller that the given switch can support or because the ratio RIF/RDF is incompatible with other ABR connections using the given switch. When a switch cannot support the values negotiated during the forward pass of the call setup, it may decide to clear the call. Additionally, the specification of the QoS class may be signaled. If the QoS class is missing, the default is class zero.

A2.2.3 Optionally Signaled Parameters

The following additional parameters can be optionally specified by the source during call setup but are optional for the source to specify. If not specified, the default value will be inserted upon call completion (without negotiation). Also, if any network element does not support Table A2.3, or a parameter in this group, the default value will be the value used.

Table A2.3 Optionally signaled ABR parameters

Parameter	Negotiation	Default Value
Nrm	no	32
Trm	no	100
CDF	up	1/16
ADTF	down	0.5

A2.2.4 Parameter Computation After Call Setup

The following parameters are computed or updated by the forward and backward sources upon completion of the call setup when FRTT and the other parameters are known.

CRM CRM is computed as: $$CRM = \left\lceil \frac{TBE}{Nrm} \right\rceil$$

ICR ICR is updated after call setup is completed to insure TBE compliance as:

$$ICR = \min\left(ICR, \frac{TBE}{FRTT} \right)$$

A2.3 RM-Cell Structure

Table A2.4 shows the fields and their position within the Resource Management (RM) cell format.

Table A2.4 Fields and their position in RM cell

FIELD	OCTET	BIT(S)	DESCRIPTION	Initial Value	
				If source-generated	If switch-generated or destination-renerated
Header	1-5	all	ATM Header	RM-VPC: VCI=6 and PTI= 110 RM-VCC: PTI=110	
ID	6	all	Protocol Identifier	1	
DIR	7	8	Direction	0	1
BN	7	7	BECN Cell	0	1
CI	7	6	Congestion Indication	0	either CI=1 or NI=1 or both
NI	7	5	No Increase	0 or 1	
RA	7	4	Request/Acknowledge	0 or set in accordance with I.371 -draft	
Reserved	7	3-1	Reserved	0	
ER	8-9	all	Explicit Cell Rate	a rate not greater than PCR parameter	Any rate value
CCR	10-11	all	Current Cell Rate	ACR parameter	0
MCR	12-13	all	Minimum Cell Rate	MCR parameter	0
QL	14-17	all	Queue Length	0 or set in accordance with I.371 -draft	
SN	18-21	all	Sequence Number	0 or set in accordance with I.371 -draft	
Reserved	22-25	all	Reserved	6A (hex) for each octet	
Reserved	52	8-3	Reserved	0	
CRC-10	52	2-1	CRC-10	See Section A2.3.1	
	53	all			

Bit	8	7	6	5	4	3	2	1
	DIR	BN	CI	NI	RA	Res.	Res.	Res.

DIR = 0 for forward RM cells
= 1 for backward RM cells
BN = 1 for Non-Source Generated (BECN) RM cells
= 0 for Source Generated RM cells
CI = 1 to indicate congestion
= 0 otherwise
NI = 1 to indicate no additive increase allowed = 0 otherwise
RA Not used for ABR. See description below

Figure A2.1 Message Type Field (Octet 7)

A2.3.1 Description of RM-cell Fields

This section describes how each field of the RM-cell is used. See Table A2.1 for requirements and options for initializing these fields. See also sections A2.5 through A2.7 for requirements and options for modifying the values in these fields.

Header: The first five bytes of an RM-cell are the standard ATM header with PTI=110 (binary) for a VCC, and additionally VCI=6 for a VPC. The CLP bit is 0 if the RM-cell is in-rate and 1 if it is out-of-rate.

ID: The protocol ID identifies the service using the RM-cell. The ITU has assigned protocol ID = 1 to ABR service.

Message Type Field

DIR: The DIR bit indicates which direction of data flow is associated with the RM-cell. A forward RM-cell, indicated by DIR=0, is associated with data cells flowing in the same direction. A backward RM-cell, indicated by DIR=1, is associated with data cells flowing in the opposite direction. DIR is changed from 0 to 1 when RM-cell is turned around at a destination.

BN: The BN bit indicates whether the RM-cell is a Backward Explicit Congestion Notification (BECN) cell (i.e., non-source generated) or not. BN=0 indicates a source generated RM-cell while BN=1 indicates a BECN RM-cell generated by a destination or a switch.

CI: The CI (congestion indication) bit allows a network element to indicate that there is congestion in the network. When a source receives a backward RM-cell with CI = 1, it decreases its ACR. When turning around a forward RM-cell, a destination will set CI = 1 to indicate that the previous received data cell had the EFCI state set.

NI: The NI (no increase) bit is used to prevent a source from increasing its ACR. In contrast to CI=1, NI=l does not require any decrease. A network element might set NI to 1 to indicate impending congestion. Normally, a source will initialize NI to 0 so that it might be allowed to increase its ACR, but it can indicate that it does not need a higher ACR by initializing NI to 1.

RA: The RA bit is not used for ATM Forum ABR.

ER: The ER (Explicit Rate) field is used to limit the source ACR to a specific value. For each RM-cell ER is set by the source to a requested rate (such as PCR). It may be subsequently reduced by any network element in the path to a value that the element can sustain. ER is formatted as a rate as defined in Section A2.3.2.

CCR: The CCR field is set by the source to its current ACR. It may be useful to network elements in computing a value to place in ER. For BECN cells, CCR=0. CCR is formatted as a rate as defined in Section A2.3.2.

MCR: The MCR field carries the connection's Minimum Cell Rate. It may be useful to network elements in allocating bandwidth among connections. For BECN cells, MCR=0. MCR is formatted as a rate as defined in Section A2.3.2.

QL: The QL field is not used for ATM Forum ABR.

SN: The SN field is not used for ATM Forum ABR.

CRC-10: The RM CRC is the same CRC used for all OAM cells. It is computed as the remainder of the division (modulo 2) by the generator polynomial of the product of x^{10} and the content of the RM-cell payload excluding the CRC field (374 bits). Each bit of this payload is considered as a coefficient (modulo 2) of a polynomial of degree 373 using the first bit as the coefficient of the highest order term. The CRC- 10 generating polynomial is: $1 + X + X^4 + X^5 + x^9 + x^{10}$. The result of the CRC calculation is placed with the least significant bit right justified in the CRC field. See ITU-T Recommendation I.610 for examples.

A2.3.2 Rate Representation

Rates in the RM-cell and in the Source Behavior are represented in a binary floating point representation employing a 5 bit exponent, *e*, a 9 bit mantissa, *m*, and a 1 bit Nonzero flag, *nz*, as described below:

$R=[2e\ (l+m/512)1*nz$ cells/seconds where,

1 bit reserved	Bit 16, most significant bit of 16 bit field
$nz \in \{0,1\}$	Bit 15 If nz = 0 the rate is zero. If nz = 1, the rate is as given by the fields *e* and *m*.
$0 < e < 31$	Bit 14 through bit 10. The mantissa is a 5 bit unsigned
$0 < m < 511$	Bit 9 through bit 1

represent all rates used in the RM-cells and source behavior for ABR service. The bit positions of a floating point rate within a 16 bit word are given below:

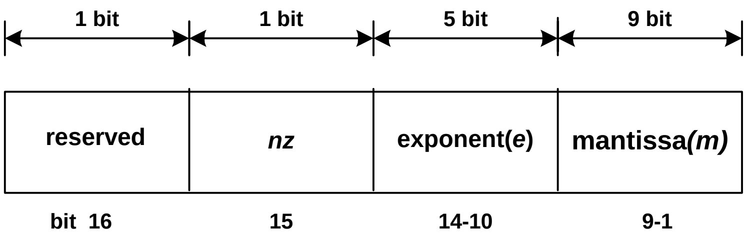


Figure A2.2 Rate format used in RM cell

Note: Bits 16-9 are transmitted before bits 8-1 when using this encoding in the RM-cell.

A2.3.3 In-rate and Out-of-rate Cell Types

ABR RM-cells shall be sent with CLP=0. ABR RM-cells with CLP=1 may be sent under the conditions explicitly stated in Sections A2.4, A2.5, and A2.6. All other ABR cells shall be sent with CLP=0. For ABR, CLP=0 cells are called "in-rate" cells, and CLP=1 cells are called "out-of-rate" cells. The use of out-of-rate RM-cells is to enable a rate increase for a connection that has an ACR of zero. The source would use the out-of-rate cells as probes to learn when it may increase its rate.

A2.4 Source Behavior

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.

1. The value of ACR shall never exceed PCR, nor shall it ever be less than MCR. The source shall never send in-rate cells at a rate exceeding ACR. The source may always send in-rate cells at a rate less than or equal to ACR.
2. Before a source sends the first cell after connection setup, it shall set ACR to at most ICR. The first in-rate cell sent shall be a forward RM-cell.
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) Nnn-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.

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

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.

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.

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.

8. When a backward RM-cell (in-rate or out-of-rate) is received with CI= l, 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 o a rate not greater than PCR. If the backward RM-cell has

NI= 1, the ACR shall not be increased.

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.

10. When generating a forward RM-cell, the source shall assign values to the various RM-cell fields as specified for source-generated cells in Table A2.4.

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.

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

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.

Notes:

1. In-rate forward and backward RM-cells are included in the source rate allocated to a connection.
2. The source is responsible for handling local congestion within its scheduler in a fair manner. This congestion occurs when the sum of the rates to be scheduled exceeds the output rate of the scheduler. The method for handling local congestion is implementation specific.

A2.5 Destination Behavior

The following items define the destination 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.

1. When a data cell is received, its EFCI indicator is saved as the EFCI state of the connection.
2. On receiving a forward RM-cell, the destination shall turn around the cell to return to the source. The

   DIR bit in the RM-cell shall be changed from "forward" to "backward", BN shall be set to and CCR, MCR, ER, CI, and NI fields in the RM-cell shall be unchanged except:

   a) If the saved EFCI state is set, then the destination shall set CI=1 in the RM-cell, and the saved EFCI state shall be reset. It is preferred that this step is performed as close to the transmission time as possible;

   b) The destination (having internal congestion) may reduce ER to whatever rate it can support and/or set CI= 1 or NI= 1. A destination shall either set the QL and SN fields to zero, preserve these fields, or set them in accordance with ITU-T Recommendation 1. 371 -draft.

   The octets defined in Table A2.4 as reserved may be set to 6A (hexadecimal) or left unchanged. The bits defined as reserved in Table A2.4 for octet 7 may be set to zero or left unchanged. The remaining fields shall be set in accordance with Section A2.3.1 (Note that this does not preclude looping fields back from the received RM-cell).
3. If a forward RM-cell is received by the destination while another turned-around RM-cell (on the same connection) is scheduled for in-rate transmission:

   a) It is recommended that the contents of the old cell are overwritten by the contents of the new cell;

   b) It is recommended that the old cell (after possibly having been over-written) shall be sent out-of-rate; alternatively the old cell may be discarded or remain scheduled for in-rate transmission;

   c) It is required that the new cell be scheduled for in-rate transmission.

4. Regardless of the alternatives chosen in destination behavior #3 above, the contents of an older cell

shall not be transmitted after the contents of a newer cell have been transmitted.

5. A destination may generate a backward RM-cell without having received a forward RM-cell. The rate of these backward RM-cells (including both in-rate and out-of-rate) shall be limited to 10 cell/second, per connection. When a destination generates an RM-cell it shall set either CI=l or NI=1, shall set BN= 1, and shall set the direction to backward. The destination shall assign values to the various RM-cell fields as specified for destination generated cells in Table A2.4.
6. When a forward RM-cell with CLP=1 is turned around it may be sent in-rate (with CLP=0) or

out-of-rate (with CLP=1).

Notes:

1. "Turn around" designates a destination process of transmitting a backward RM-cell in response to having received a forward RM-cell.
2. It is recommended to turn around as many RM-cells as possible to minimize turnaround delay, first by using in-rate opportunities and then by using out-of-rate opportunities as available.

A2.6 Switch Behavior

The following items define the switch behavior for CLP=0 and CLP=1cell 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.

1. A switch shall implement at least one of the following methods to control congestion at queuing

   points:

   a) *EFCI marking:* The switch may set the EFCI state in the data cell headers
   b) *Relative Rate Marking:* The switch may set CI=1 or NI=1 in forward and/or backward RM-cells
   c) *Explicit Rate Marking:* The switch may reduce the ER field of forward and/or backward RM-cells (Explicit Rate Marking)
   d) *VS/VD Control:* The switch may segment the ABR control loop using a virtual source and destination.
2. A switch may generate a backward RM-cell. The rate of these backward RM-cells (including both in-rate and out-of-rate) shall be limited to 10 cells/second, per connection. When a switch generates a RM-cell it shall set either CI=1 or NI=1, shall set BN=1, and

shall set the direction to backward. The switch shall assign values to the various RM-cell fields as specified for switch-generated cells in Table A2.4.

3. RM-cells may be transmitted out of sequence with respect to data cells. Sequence integrity within the RM-cell stream must be maintained.
4. For RM-cells that transit a switch (i.e., are received and then forwarded), the values of the various fields before the CRC-10 shall be unchanged except:
   a) CI, NI, and ER may be modified as noted in #1 above
   b) RA, QL,and SN may be set in accordance with ITU-T RecommendationI.371-draft
   c) MCR may be corrected to the connection's MCR if the incoming MCR value is incorrect.
5. The switch may implement a use-it-or-lose-it policy to reduce an ACR to a value which approximates the actual cell transmission rate from the source.

Notes:

1. A switch queuing point is a point of switch contention where cells may be potentially delayed or lost. A switch may contain multiple queuing points.
2. The implications of combinations of the above methods is beyond the scope of this specification.

A2.7 Virtual Source and Virtual Destination Behavior

VS/VD behavior divides an ABR connection into two or more separately controlled ABR segments. The coupling between adjacent ABR control segments associated with an ABR connection is implementation specific. Figure A2.5 illustrates an ABR virtual connection which incorporates segmentation.

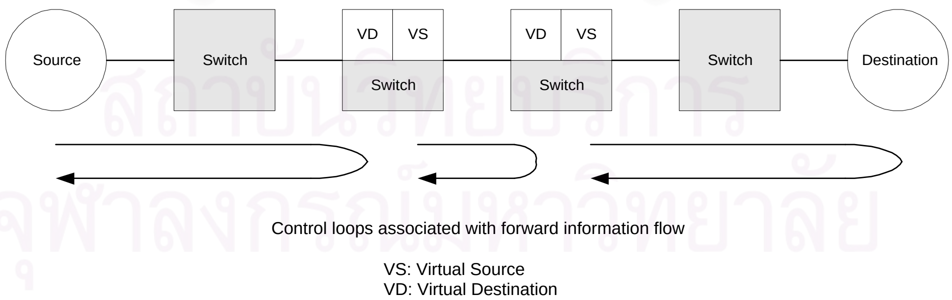


Figure A2.3 Example of a segmented ABR virtual connection

The following applies to VS/VD behavior:

1. Each ABR control segment, except the first, is sourced by a virtual source. A virtual source assumes the behavior of an ABR source end point. Backward RM-cells received by a virtual source are removed from the connection.
2. Each ABR control segment, except the last, is terminated by a virtual destination. A virtual destination assumes the behavior of an ABR destination end point. Forward RM-cells received by a virtual destination shall be turned around as defined in destination behavior #2, and shall not be forwarded to the next segment of the connection.
3. The coupling between two adjacent ABR control segments associated with an ABR connection is implementation specific.
4. MCR shall be conveyed across VS/VD boundaries.
5. Setting of other parameters at VS/VD is network specific.

A2.8 Point-to-Multipoint Behavior

The support of ABR point-to-multipoint connections is not required for ABR compliance as defined in this specification. However, the guidelines provided here are intended as a basic framework for a complete specification of point-to-multipoint ABR service in the future.

The operation of an ABR point-to-multipoint connection is functionally divided into behaviors for ABR sources/virtual sources, destinations/virtual destinations, switches, and branch points. According to their functional definitions, a source and destination is located at the root of the point-to-multipoint tree and at each of the leaves,

- one or more virtual sources/virtual destinations may be located on each branch of the tree,
- one or more switches may be located on each branch of the tree, and
- a branch-point is located at the intersection of two or more branches.

Note that switches, which determine the feedback sent from queuing points, are considered as functionally separate from branch points, which replicate cells traveling from root to leaves and consolidate feedback traveling from leaves to root. A branch is defined as any point-to-point segment of the point-to-multipoint
tree. A branch may be classified as in the "non-responding state" if it has not transmitted (e.g., turned
around) RM-cells towards the root for a time, the length of which is network specific but indicates the
unavailability of the branch. Otherwise, the branch is in the "responding state." The classification of a branch as in the "non-responding state" is optional.

A2.8.1 Behavior for Sources, Destinations, Switches, and VS/VD of Point-to-Multipoint Connections

For a point-to-multipoint connection, the source behavior is the same as in Section A2.4, except that data cells shall not be transmitted in the direction from the leaves to the root. The destination behavior is the same as in Section A2.5, the switch behavior the same as in Section A2.6, and the virtual source/virtual destination behavior the same in Section A2.7

A2.8.2 Behavior for Branch Points on Point-to-Multipoint Connections

1. An ABR branch point shall replicate each data cell and RM-cell receive from the root onto each branch that leads to a leaf, whenever the branch is in the responding state. RM-cells may be transmitted onto the leaves out of sequence with respect to data cells, but the sequence integrity within the RM-cell stream transmitted to each branch must be preserved.
2. An ABR branch point shall transmit forward and backward RM-cells towards the root. This may be done by consolidating the information from forward and backward RM-cells received in the leaf-to-root direction from each branch in the responding state. However, a branch point is responsible for assuring that the ABR flow transmitted to each branch (both towards the leaves and towards the root) conforms to the expected behavior for a point-to-point ABR flow, given that the ABR flows received by the branch point do. There are network elements which support traditional multicasting (i.e. cell duplication) but can not consolidate RM-cells. Their role within the framework of point-to-multipoint ABR service requires further study.
3. An ABR branch point may:
    - buffer data and generate ABR feedback from queuing points as defined by the switch behavior in Section A2.6;
    - implement virtual sources and virtual destinations at one or more branches as defined in Section A2.7.

VITA

Mr. Naris Rangsinoppamas received his B.Eng. in electrical engineering from King Mongkut’s Institute of Technology North Bangkok in 1991 and M.Eng. from Chulalongkorn University in 1995. From 1992 to 1995 he worked at KMITNB as an instructor. Then he joined Nokia Telecommunications as a transmission engineer until 1996. His research interests are ATM traffic management, and teletraffic and mobile communiations. He is a student member of IEICE.