Fuzzy Leaky Bucket Congestion Control in ATM Networks

by

Jianqin Weng

A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfillment of
the requirements for the degree of
Master of Engineering

Ottawa-Carleton Institute for Electrical Engineering
Faculty of Engineering
Department of Systems and Computer Engineering
Carleton University,
1125 Colonel Drive
Ottawa, Ontario, Canada
K1S 5B6

December 4, 1996

© copyright
1996, Jianqin Weng

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

—

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

—

ISBN 0-612-19525-2

Electronics and Electrical

UMI

Ancient	0579
Medieval	0581
Modern	0582
Black	0028
African	0031
Asia, Australia and Oceania	0032
Canadian	0034
European	0035
Latin American	0036
Middle Eastern	0033
United States	0037
History of Science	0583

Law	0398
Political Science	
General	0415
International Law and	
Relations	0414
Public Administration	0417
Recreation	0814
Work	0452
Sociology	
General	0426
Criminology and Penology	0427
Demography	0938
Economic and Capital Studies	0431
Marriage and Family	
Studies	0428
Industrial and Labor	
Relations	0429
Religion and Social Welfare	0430
Social Structure and	
Development	0430
Theory and Methods	0344
Transportation	0709
Urban and Regional Planning	0999
Women's Studies	0453

Engineering	0.53
General	0.53
Aerospace	0.53
Agriculture	0.53
Automotive	0.54
Biomedical	0.54
Chemical	0.54
Civil	0.54
Electrical and Electronic	0.54
Heat and Thermodynamics	0.54
Hydraulic	0.54
Industrial	0.54
Materials	0.54
Mathematical Sciences	0.79
Mechanical	0.79
Metallurgy	0.72
Mining	0.55
Naval	0.55
Packaging	0.54
Petroleum	0.75
Sanitary and Municipal	0.55
System Sciences	0.79
General	0.79
Operations	0.79
Planning	0.79
Plastic Technology	0.79
Textile Technology	0.79
PSYCHOLOGY	
General	0.42
Behavioral	0.38
Clinical	0.42
Developmental	0.42
Experimental	0.42
Industrial	0.42
Personality	0.42
Physiological	0.42
Psychiatric	0.42
Psychosomatic	0.42
Social	0.42

The undersigned hereby recommend to
the Faculty of Graduate Studies and Research
acceptance of the thesis

Fuzzy Leaky Bucket Congestion Control in ATM Networks

submitted by

Jianqin Weng, B.Eng

in partial fulfillment of the requirements for the degree of Master of Engineering

Thesis Supervisor

Chair, Department of Systems and Computer Engineering

Carleton University
December 4, 1996

Abstract

This thesis proposes an ATM congestion control mechanism that introduces a leaky bucket control scheme based on fuzzy logic principles. Network congestion is described linguistically by introducing a fuzzy rule and appropriate fuzzy variables, and is treated mathematically via fuzzy set manipulations. With the application of fuzzy logic the complex mathematical treatment of classical feedback control is avoided, and the “hard” bound decision effect in the traditional LB is also eliminated in favor of “soft” bound membership functions.

In order to evaluate the effectiveness of the fuzzy LB, the performance of the ATM network with a non-fuzzy adaptive LB mechanism is compared. Network parameters which affect the performance are identified and optimized for the two control schemes and a comparison of the two approaches is carried out under the same network condition and optimal parameters. Finally, the performance of the fuzzy LB and adaptive LB are also evaluated under self-similar traffic loading condition.

The performance analysis in this paper is based on simulation combined with numerical optimization methods. Our results indicate that the fuzzy leaky bucket mechanism leads to significant improvement to the system performance.

Acknowledgements

First and foremost, I wish to express my profound gratitude to my supervisor, Professor I. Lambadaris, for his invaluable guidance, inspiration, encouragement and assistance throughout the span of this research. His wide knowledge, and hard working attitude stimulated me to learn more and do more.

I would like to thank Professor Devetsikiotis for reviewing this thesis and give valuable comments and suggestions.

The support and assistance from Professor R. Kaye and the other professors in Department of Systems and Computer Engineering are also greatly appreciated.

Finally, I would like to express my deep appreciation to my husband Ke-qin, my lovely daughter Jennifer, and my family members in China, without whose love, encouragement, and support this work would not have been possible.

List of Acronyms

AI	Artificial Intelligence
ATM	Asynchronous Transfer Mode
B-ISDN	Broadband Integrated Service Digital Network
CCITT	Consultative Committee for International Telegraphy and Telephony
F-ARIMA	Fractional Autoregressive Integrated Moving-Average
FGN	Fractional Gaussian Noise
FLC	Fuzzy Logic Controller
ISDN	Integrated Services Digital Network
LB	Leaky Bucket
LAN	Local Area Network
LRD	Long Range Dependence
MFA	Mean Field Annealing
QoS	Quality of Service
STM	Synchronous Transfer Mode
VC	Virtual Channel
VLB	Virtual Leaky Bucket
VP	Virtual Path

List of Symbols

α_1	Decrease Rate of Token Generation Interval
α_2	Increase Rate of Token Generation Interval
C	Lagrangian Cost Function
D	Cell Queueing Delay
H	Hurst Parameter
k_1	Slope of “Small” Membership Function
k_2	Slope of “Large” Membership Function
L	Ratio of Cell Loss
L_p	Ratio of Cell Loss Criteria
R	Replication
θ_1	Low Threshold of Network Buffer Occupancy
θ_2	High Threshold of Network Buffer Occupancy

Table of Contents

Abstract	i
List of Acronyms	iii
List of Symbols	iv
Table of Contents	v
List of Figures	viii
List of Tables	x
Chapter 1 Introduction	1
1.1 Background	1
1.2 Motivation	2
1.3 Objectives of the Thesis	5
1.4 Thesis Organization	6
Chapter 2 Fuzzy Logic Controller in ATM Networks	7
2.1 Introduction	7
2.2 Fuzzy Set and Fuzzy Logic	8
2.2.1 Fuzzy Terminology	8
2.2.2 Fuzzy Set Theoretic Operations	11
2.2.3 Fuzzy (or Linguistic) Variables	12
2.2.4 Fuzzy Implication	13
2.3 Fuzzy Logic Controller	14
Chapter 3 Adaptive Leaky Bucket and Fuzzy Leaky Bucket	17
3.1 Adaptive Leaky Bucket	17
3.2 Fuzzy Leaky Bucket Mechanism	18
Chapter 4 Network Performance Evaluation	23
4.1 Introduction	23

4.2	Network Simulation	23
4.2.1	Network Topology	23
4.2.2	Quality of Service Parameters	24
4.3	Simulation Model and Assumptions.....	25
4.3.1	Simulation Model	25
4.3.2	Poisson Process Traffic Model.....	26
4.3.3	Self-Similar Traffic Model.....	27
4.3.4	Video Trace	31
4.3.5	Transient Time and Variance Reduction	32
4.3.6	Assumptions for the Network Parameters	33
4.4	Preliminary Results and Methodology for Adaptive LB.....	34
4.4.1	Performance Measures.....	34
4.4.2	Cell Loss Criteria L_p	36
4.5	Simulation Results of Adaptive LB	38
4.5.1	Design Optimization for Adaptive LB.....	38
4.5.2	Adaptive LB Simulation Results Using Poisson Batch Arrival Process for Data Sources.....	39
4.5.3	Adaptive LB Simulation Results Using Self-Similar Data Traffic....	40
4.6	Fuzzy LB Controller	42
4.6.1	Implementation of the Fuzzy LB Control	42
4.6.2	Cell Loss Characterization by Applying Fuzzy LB	46
4.7	Fuzzy LB Simulation Results	46
4.7.1	Fuzzy LB Simulation Results Using Poisson Process Data Traffic...	46
4.7.2	Fuzzy LB Simulation Results Using Self-Similar Data Traffic	48
4.8	Comparison of the Two LB Control Schemes	50
Chapter 5	Conclusions and Suggestions for Further Study	53
5.1	Summary and Conclusions	53
5.2	Suggestions for Further Investigations	55

5.2.1	Expansion of Control Parameter Space	55
5.2.2	Video Rate Control	55
References		56

List of Figures

Figure 2-1	Representation of fuzzy speeds.....	13
Figure 2-2	Fuzzy logic controller	14
Figure 3-1	Fuzzy leaky bucket	19
Figure 3-2	Partitioning of the buffer occupancy corresponding to two thresholds	20
Figure 3-3	Modified buffer occupancy corresponding to three fuzzy states	20
Figure 3-4	Fuzzy variable membership functions	21
Figure 3-5	Fuzzy reasoning and implication of fuzzy LB	22
Figure 4-1	ATM network topology	24
Figure 4-2	ATM simulation queueing model.....	26
Figure 4-3	Video frame empirical distribution	30
Figure 4-4	Bit rate of video traffic	31
Figure 4-5	Estimated mean queueing delay vs. simulation run time.....	32
Figure 4-6	Cell loss $L(\theta_1, \theta_2)$ using the Poisson process for data sources ($\rho = 80\%$).....	36
Figure 4-7	Total mean queueing delay $D(\theta_1, \theta_2)$ using the Poisson process for data sources ($\rho = 65\%$).....	36
Figure 4-8	(θ_1, θ_2) trace recorded by running MFA.....	38
Figure 4-9	Queueing delay in adaptive control using Poisson data sources.....	40
Figure 4-10	Queueing delay in adaptive control using self-similar data sources with different hurst parameters.....	42
Figure 4-11	Network buffer occupancy as determined by optimal parameters.....	44
Figure 4-12	Modified network buffer occupancy.	44
Figure 4-13	Membership function of token generation interval.....	45
Figure 4-14	Queueing delay in fuzzy control using Poisson data sources.	47

Figure 4-15	Queueing delay in fuzzy control using self-similar data sources with different hurst parameter.....	49
Figure 4-16	Comparison of the queueing delay (Poisson arrival process for data sources).....	51
Figure 4-17	Comparison of the queueing delay (self-similar traffic $H=0.6$).....	51
Figure 4-18	Comparison of the queueing delay (self-similar traffic $H=0.75$).....	52
Figure 4-19	Comparison of the queueing delay (self-similar traffic $H=0.95$).....	52

List of Tables

Table 2-1	Temperature versus crisp logic value.....	9
Table 2-2	Temperature versus fuzzy logic value.....	10
Table 4-1	Network traffic load vs. average Poisson batch arrival rate.....	27
Table 4-2	Network parameter assumptions.....	34
Table 4-3	MFA range of values for (θ_1, θ_2)	37
Table 4-4	Minimum cell loss ratio in adaptive LB	37
Table 4-5	Network traffic load vs. optimal parameters in adaptive control corresponding to Poisson batch arrival process for data sources	39
Table 4-6	Network traffic load vs. optimal parameters in adaptive control corresponding to self-similar data sources with Hurst parameter = 0.6	41
Table 4-7	Network traffic load vs. optimal parameters in adaptive control corresponding to self-similar data sources with Hurst parameter = 0.75	41
Table 4-8	Network traffic load vs. optimal parameters in adaptive control corresponding to self-similar data sources with Hurst parameter = 0.95	41
Table 4-9	range of values for the analysis of the fuzzy LB control	45
Table 4-10	Minimum cell loss ratio in fuzzy LB	46
Table 4-11	Network traffic load vs. optimal parameter in fuzzy control corresponding to Poisson data sources	47
Table 4-12	Network traffic load vs. optimal parameters in fuzzy control corresponding to self-similar data sources with Hurst parameter = 0.6	48
Table 4-13	Network traffic load vs. optimal parameters in fuzzy control corresponding to self-similar data sources with Hurst parameter = 0.75	48

Table 4-14	Network traffic load vs. optimal parameters in fuzzy control corresponding to self-similar data sources with Hurst parameter = 0.95	49
------------	--	----

Chapter 1 Introduction

1.1 Background

The emergence of high-speed telecommunications technologies has revolutionized the communications industry and has created the possibility of development the Broadband Integrated Services Digital Network (B-ISDN) [33]. The B-ISDN is conceived as an all-purpose digital network, which enhances the functions of the existing networks to provide an integrated access that will support a variety of applications for its customers in a flexible and cost-effective manner[13][18].

The transfer mode of the broadband network, as recommended by the International Consultative Committee for Telephone and Telegraph (CCITT) on ISDN broadband aspects, is Asynchronous Transfer Mode (ATM)[12][27].

Unlike traditional data networks, ATM networks will be required to carry a broad range of traffic classes ranging from bursty, variable-rate sources, such as voice and video, to smooth, constant bit rate sources [7] [18]. Moreover, they will have to provide a guaranteed performance or quality of service (QoS) to many of these traffic classes. The ability to guarantee performance is particularly important in networks for supporting real time applications. The criteria of QoS performance is likely to vary from one application to another, but are projected to include such measures as cell loss, delay, and delay jitter guarantees. Therefore an ATM networks need to provide end-to-end QoS guarantees while still taking advantage of the burstiness of the traffic via the use of statistical multiplexing.

ATM is a packet-oriented switching and multiplexing technique based on the use of fixed length cells and is expected to offer high bit rate flexibility and a great degree of inte-

gration, in both transmission and switching functions [20] [34]. It will not allocate fixed channel bandwidth to individual connections. Rather, multiple logic connections will be statistically multiplexed over the same physical link so that each connection has access to the full available bandwidth “on demand”.

ATM shares much in common with conventional packet switching at lower rates, in which the subject of traffic and congestion control has been studied extensively and for which several workable control techniques are known (e.g., window control protocol). However, most of these conventional approaches to traffic and congestion control do not scale well to the extremely high speeds associated with ATM networks [8] [35]. An architecture for traffic and congestion control which offers simplicity, flexibility, robustness, and controllability to effectively meet the need of an ATM environment is highly demanded. The major challenge now is to successfully design and employ the ATM traffic control.

1.2 Motivation

Generally, the traffic control in ATM networks fall into two major categories: preventive and reactive [24]. Preventive controls take actions to prevent congestion from occurring, it can be performed in two ways: admission control and bandwidth enforcement [15]. Reactive controls take actions to recover from congestion once it occurs and try to bring the degree of network congestion to an acceptable level.

The LB mechanism is a preventive control policing algorithm[31] [32], which ensure that sources do not exceed allocated parameters once a connection has been admitted to the network. In the leaky bucket mechanism, a cell is accepted only when it can draw a token from a token pool. If the token pool is empty, then the cell is discarded. Tokens are generated at a given rate and stored in the token pool. The pool is supposed to be of finite size M . If the token pool is full, additional tokens are discarded. M can be seen

as the maximum allowable burst length since a maximum of M cells may be transmitted at one time.

This traffic policing LB algorithm can be combined with traffic shaping in a system in which incoming cells can be queued in the cell buffer instead of being discarded when the token pool is empty[2]. The cell blocking probability, the probability that a cell arrives to an empty token pool, depends on the size of cell buffer and token pool. The purpose of a cell buffer is to handle a system in which the token pool control can turn on only during times of congestion.

The drawback of LB [31] is the control parameters must be set with a large tolerance in order to guarantee the contracted mean rate. Otherwise, incoming cells can be lost even though the long term average rate of the source is within the allocated bandwidth.

A Virtual Leaky Bucket (VLB), has been proposed [11] [38] in order to overcome the limitations of the LB policing method. Cells arriving at an empty token pool are marked and transmitted without a token. Marked cells are considered violators of allocated bandwidth. If at some point along its path a marked cell reaches a congestion link, it may be discarded so the throughput of the unmarked cells is not significantly affected. This not only allows us to take advantage of a light network load, but also enables a larger margin of error in determining the token pool parameters. But when traffic sources are very bursty, the VLB mechanism may frequently cause the network congestion, and therefore more cell loss since a large number of violated cells are marked and allowed to enter the network. The performance requirements can hardly be met.

Due to the bursty and unpredictable nature of traffic, variable Bit Rate (VBR) traffic could produce temporary overloads at some internal network buffers. In such case, the above mentioned preventive LB control approaches are not sufficient to control the con-

gestion, and additive reactive controls may be necessary to help the congested portions of a broadband network recover to their normal state.

A notable example is combined reactive/preventive LB approach[26]. In this approach, the authors propose an adaptive rate based congestion control mechanism based on the LB scheme. The parameters of the mechanism are controlled using feedback information sent by the network. The network is modeled as a multiplexer fed by a number of sources each of which is controlled by a leaky bucket. A simple threshold policy at the multiplexer queue is used to detect the congestion.

Another similar LB approach to congestion control was based on the two thresholds control method called adaptive LB [37]. In this proposal, two threshold values were used to determine the degree of network congestion, and LB token generation interval is changed during the congestion period.

However, all of these congestion control schemes that utilize either buffer threshold or capacity estimation, suffer from some fundamental limitations. Generally, it is difficult for a network to acquire complete statistics of input traffic. As a result, it is not easy to accurately determine the effective thresholds in the ATM networks [5]. Therefore, a network is forced to make a decision based on incomplete information. This is the motivation behind developing a controller having a greater ability to adapt to dynamic, imprecise, and bursty ATM environments without using hard bound thresholds.

Recently, fuzzy logic system have been widely applied to control nonlinear and time-varying systems in which they can provide simple and effective solutions [3].

In [28] Ndousse proposed a fuzzy controller for managing cell generated by voice sources in ATM networks. This is a fuzzy logic implementation of Virtual Leaky Bucket. Bonde and Ghost [4] introduced fuzzy mathematics to provide flexible and high performance solution to queue management in ATM networks. The notion of the cell blocking,

wherein a fuzzy threshold function, based on Zadeh's fuzzy set theory [23], is utilized to deliberately refuse entry to the fraction of incoming cells from other switches. The blocked cells are then rerouted to other switches. Recently, Cheng [5] proposed a fuzzy traffic controller that can simultaneously manage call admission control and congestion control for ATM network. This paper introduces a performance analysis methodology which does not involve parameter optimization. In contrast, we develop an optimization method to select the network parameters for optimal performance.

1.3 Objectives of the Thesis

Overall, previous research has provided strong indications that fuzzy set theory can provide a robust mathematical framework to deal with real world imprecision. In this thesis, we propose an ATM congestion control mechanism that introduces a leaky bucket scheme based on fuzzy logic principles. The aim of this approach is to overcome the limitations found in traditional leaky bucket control scheme and present a new and simple methodology to deal with an ATM network congestion. The fuzzy LB controller that we will present in this study, processes the feedback information (indicating the network congestion status) using a mathematical approach based on fuzzy set manipulation. By employing appropriate fuzzy rules and fuzzy membership functions, the "threshold" of the network congestion status is assigned a continuous membership function which covers a specified range of buffer occupancy values. In this way each network occupancy level yields a different token generation rate. The main objectives of the thesis are as follows.

- 1) Develop a fuzzy Leaky Bucket (LB) congestion control scheme based on fuzzy logic principles. The leaky bucket parameters such as token generation interval may be a function of the network load or congestion. By formulating a set of fuzzy rules between network congestion and LB token generation interval, the complex mathematical treatment of the classical feedback control is avoided, and the hard bound membership effect in the traditional LB is also eliminated in favor of "soft" bound membership functions.

2) Identify both fuzzy LB parameters and traditional adaptive LB parameters that affect the effectiveness of the congestion control strategies. Optimize both approaches in order to get maximum performance.

3) Compare the optimal fuzzy LB and the adaptive LB schemes by developing two ATM network simulations. The first is an ATM network simulation using adaptive LB control scheme. Another is ATM network simulation using fuzzy LB control scheme. The performance of fuzzy LB control scheme and adaptive LB control scheme is evaluated under the same network condition and (optimal) parameter setting. Finally, the performance of the fuzzy LB and adaptive LB is also evaluated under self-similar traffic load [16].

1.4 Thesis Organization

The remaining chapters of this thesis are organized as follows:

In Chapter 2 we review the fundamentals of fuzzy logic, fuzzy sets, fuzzy set operation, and fuzzy logic controller. The difference between crisp set and fuzzy set is also compared. Fuzzy logic controller algorithm is introduced.

In Chapter 3 the fuzzy LB control scheme including fuzzy rules and membership functions is developed. The adaptive leaky bucket control scheme is also introduced.

In Chapter 4 we present our simulation methodology and discuss our results of the comparison between the two LB controllers.

In Chapter 5 conclusions and suggestions for further work are given.

Chapter 2 Fuzzy Logic Controller in ATM Networks

2.1 Introduction

One of the latest control methodologies is based on the so called “intelligent control” which is defined as a combination of control theory and Artificial Intelligence (AI) [3] [22]. Among many possible new methodologies based on AI, fuzzy logic is a popular area which has received a considerable research interest in recent years.

Fuzzy logic comprises a set of principles on which fuzzy control is based. Fuzzy control is much closer in principle to human thinking and natural language compared to the traditional logic systems [19] [22]. Unlike binary (yes or no) information, fuzzy logic is a multivalued logic that allows for degrees (e.g., normal versus slow or fast) of set membership, therefore, it may provide an effective way of capturing the approximate, inexact nature of the real world.

A significant number of commercial applications rely on fuzzy logic. Examples include as controller for tasks such as managing temperatures and energy efficiency in heating and cooling devices and regulating timing and fuel flow in automobile engines [9]. Controllers also are used to make constant operating adjustments to subway trains, home appliances, cameras, and elevators. In this thesis, it will be used in ATM network congestion control [9].

Fuzzy logic's benefits are fault tolerance and the ability to provide accurate responses to ambiguous data. A Fuzzy Logic Controller (FLC) provides an algorithm

which converts the linguistic control strategies based on expert information into an automatic control strategy. The methodology of the FLC appears very useful when the processes are too complex for analysis by conventional quantitative techniques, and attractive when the available sources of information are interpreted qualitatively, inexactly, or uncertainly.

Since ATM networks form an appropriate transport for a variety of types of traffic, the ATM traffic controller has to take into account the different traffic characteristics (e.g., burstiness, peak rate, and average bit rate) in order to guarantee certain quality of service requirements. Since there is a dynamic change of the network traffic, it is very difficult to get an analytical description matching the network node situations and modelling burst traffic sources. A good controller is required to solve this complex problem. Fuzzy logic has been applied for admission control seems to be a promising method for traffic shaping in ATM networks [17].

The purpose of this chapter is to introduce the concept of fuzzy theory and fuzzy logic control. The principles will be applied in the next chapter for the development of a fuzzy ATM traffic controller.

2.2 Fuzzy Set and Fuzzy Logic

The fuzzy set theory is a generalization of conventional set theory that was introduced by Zadeh in 1965 as a mathematical way to represent vagueness in everyday life[3].

2.2.1 Fuzzy Terminology

Brief descriptions of several major terminologies of fuzzy logic are presented in what follows [19].

- **Crisp Logic:** Another name for traditional Boolean logic to differentiate it from fuzzy logic. In crisp logic, the three logic operations AND, OR, and NOT return either 1 or 0.
- **Crisp Set:** The traditional definition of a set in classical logic. Crisp sets have strict membership criteria according to which an object is either completely included or excluded from the set. They are mathematical sets with definitive boundaries. For example, the oil temperature is between 300°F to 500°F. In crisp logic, two logic values can be used to describe the oil temperature. Assume that the oil temperature being higher than 400°F refers hot oil (corresponding to crisp logic 1) and the oil temperature being lower than 400°F corresponds to cold oil (corresponding to crisp logic 0). The mappings of temperature and crisp logic value are shown in Table 2-1.

The set of hot oil = {temperature between 400°F - 500°F}

The set of cold oil = {temperature between 300°F - 400°F}

Table 2-1 Temperature versus crisp logic value

Temperature (°F)	Crisp logic value	Meaning
320	0	cold oil
360	0	cold oil
380	0	cold oil
420	1	hot oil
460	1	hot oil
500	1	hot oil

- **Fuzzy Logic:** The process of solving problems that deal with ambiguous data using a multivalued logic to represent a crisp-logic system. Fuzzy logic holds that all things are a matter of degree (e.g., warm versus cold or hot). In fuzzy logic, the three logic operations AND, OR, and NOT return a degree of membership that is a num-

ber valued between 0 and 1. We still use the previous oil temperature example. In fuzzy logic, the multiple logic value can be used to describe the oil temperature. Assume that the oil temperature 400°F means warm oil, the oil temperature 500°F means very hot oil, and the oil temperature 320°F is very cold oil. The possible mapping of temperature and fuzzy logic value is shown in Table 2-2.

Table 2-2 Temperature versus fuzzy logic value

temperature (°F)	Fuzzy logic value	Meaning
320	0.1	very cold oil
360	0.3	cold oil
380	0.4	slight warm oil
420	0.6	very warm oil
460	0.8	hot oil
500	1	very hot oil

- **Fuzzification:** The process for converting crisp numerical input to fuzzy logic value by combining actual input values with stored membership functions. For the example shown in Table 2-2, the input of controller is temperature. After fuzzification the temperature is converted to a fuzzy logic value.
- **Fuzzy Inference:** Procedure for processing the input fuzzy logic value according to the fuzzy control rules, and yielding a control output (fuzzy logic value).
- **Defuzzification:** A process in which a fuzzy output is converted into a crisp, numerical result.
- **Fuzzy Set:** A nontraditional type of set that allows an element to have gradual or partial degree of membership. In fuzzy logic, the traditional Boolean values of true and false (1 and 0) are replaced by continuous-set membership values ranging from 0 to 1. In the previous oil temperature example, assume the fuzzy set $A = \{\text{hot oil}\}$.

then temperature 320°F has a 0.1 degree of membership in A, temperature 400°F has a 0.5 degree of membership in A, while temperature 480°F has a 0.9 degree in A.

- **Membership:** The degree of inclusion in a set such as 0.1, 0.5, and 0.9 in the above example. Fuzzy sets have values between 0 and 1 that indicate the degree to which an element has membership in the set. At 0, the element has no membership; while at 1, it has full membership.

2.2.2 Fuzzy Set Theoretic Operations

Let U be a set $U = \{x_1, x_2, x_3, \dots\}$, where x_i could be discrete or continuous. U is called the universe of discourse[22]. x_i represents a generic element of U . Let A and B be two fuzzy sets in U with membership functions μ_A and μ_B , respectively. The set theoretic operations of union, intersection and complement for fuzzy sets are defined via their membership functions. We now briefly outline the definitions of the operations used in the thesis[22].

Definition 1 Union: The membership function $\mu_{A \cup B}$ of the union $A \cup B$ is defined for all $x \in U$ by

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) = \max\{\mu_A(x), \mu_B(x)\} \quad (2.2.2.1)$$

Definition 2 Intersection: The membership function $\mu_{A \cap B}$ of the intersection $A \cap B$ is defined for all $x \in U$ by

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) = \min\{\mu_A(x), \mu_B(x)\} \quad (2.2.2.2)$$

Definition 3 Complement: The membership function $\mu_{\bar{A}}$ of the complement of a fuzzy set A is defined for all $x \in U$ by

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (2.2.2.3)$$

Definition 4 Cartesian Product: If $A_1, \dots, A_n$ are fuzzy sets in $U_1, \dots, U_n$, respectively, the Cartesian product of $A_1, \dots, A_n$ is a fuzzy set in the product space $U_1 \times \dots \times U_n$ with membership function defined as:

$$\mu_{A_1 \times \dots \times A_n}(x_1, x_2, \dots, x_n) = \min \{ \mu_{A_1}(x_1), \dots, \mu_{A_n}(x_n) \} \quad (2.2.2.4)$$

Definition 5 Fuzzy relation: An n-ary fuzzy relation is a fuzzy set in $U_1 \times \dots \times U_n$ and is expressed as

$$R_{U_1 \times \dots \times U_n} = \{ [(x_1, \dots, x_n), \mu_R(x_1, \dots, x_n)] \mid (x_1, \dots, x_n) \in U_1 \times \dots \times U_n \} \quad (2.2.2.5)$$

2.2.3 Fuzzy (or Linguistic) Variables

The use of fuzzy sets provides a basis of a systematic way for the manipulation of vague and imprecise concepts. In particular, we can employ fuzzy sets to represent linguistic variables. A fuzzy variable can be regarded as a variable whose values are defined in linguistic terms. More specifically:

Definition 6 Fuzzy Variables: A fuzzy variable is characterized by a 3-tuple $(x, T(x), U)$, where x is the name of variable. $T(x)$ is referred to as the term set of x in the fuzzy logic literature[22]. U is the universe of discourse. We now present an example to explain the meaning of a fuzzy variable. Let us assume that *speed* is interpreted as a fuzzy variable, then its term set $T(\text{speed})$ could be

$$T(\text{speed}) = \{\text{slow}, \text{moderate}, \text{fast}\}$$

where each term in $T(\text{speed})$ is characterized by a fuzzy set in a universe of discourse $U = [0, 10]$. We might interpret “slow” as “a speed below about 4mph,” “moderate” as “a speed close to 5.5mph,” and “fast” as “a speed above 7mph.” These terms can be characterized as fuzzy sets whose membership functions “slow,” “moderate” and “fast” are shown in Figure 2-1.

Figure 2-1 Representation of fuzzy speeds

2.2.4 Fuzzy Implication

A fuzzy control rule such as “if x is A then y is B ” is represented by a fuzzy implication function and is denoted by $A \rightarrow B$, where A and B are fuzzy set in universes U and V with membership function $\mu_A(x)$ and $\mu_B(y)$, respectively.

Definition 7 Fuzzy Conjunction: The fuzzy conjunction is defined [23] for all $x \in U$ and $y \in V$ by

$$A \rightarrow B = A \times B = \int_{U \times V} (\mu_A(x) \bullet \mu_B(y)) d(x, y) \quad (2.2.4.1)$$

where $\bullet$ is an operator such as union or intersection.

Mamdani's mini-fuzzy implication [23]:

Mamdani's mini-fuzzy implication R_c is defined by using the definition of the fuzzy conjunction if an intersection operator is used in (2.2.4.1).

$$R_c = A \times B = \int_{U \times V} (\mu_A(x) \wedge \mu_B(y)) d(x, y) \quad (2.2.4.2)$$

If the input is fuzzy singleton x_0 , then the results derived by employing Mamdani's minimum operation rule R_c can be expressed simply as

$$R_c: \hat{\alpha}_i \wedge \mu_{B_i}(y) \quad (2.2.4.3)$$

where $\hat{\alpha}_i = \mu_{A_i}(x_0)$

2.3 Fuzzy Logic Controller

As mentioned earlier, a fuzzy logic controller consists of three stages (or processes), fuzzification, inference engine and defuzzification as shown in Figure 2-2. The controller

Figure 2-2 Fuzzy logic controller

receives a crisp input x , and then it “fuzzifies” the crisp values in the fuzzification stage. In the inference engine stage, according to a set of rules that have been selected either on intuition or according to the desired response of the system, each rule’s consequent then contributes to the final result to the degree that its antecedent is true. A defuzzification process transforms the fuzzy value back to crisp value to provide the final control result.

The functions of each stage are now described in detail. The fuzzification stage involves the following steps:

- a) measurement of the value of input variables.
- b) implementation of a scale mapping of the input value if necessary.
- c) conversion of input value into suitable fuzzy values.

The inference engine is a processor that uses the degree of truth in rule premises to select an appropriate rule to execute. It has the capability of simulating human decision-making based on fuzzy concepts and of inferring fuzzy control actions employing fuzzy rules of inference. It is a way of producing fuzzy responses from fuzzy rules. A fuzzy rule is defined by the relation between condition (or antecedent) and action (or consequent).

If (a set of conditions are satisfied) THEN (a set of consequences can be inferred).

A fuzzy system is characterized by a set of linguistic description rules based on expert knowledge. In the case of single-input-single-output fuzzy systems, fuzzy control rules can be expressed as:

R1: If x is A_1 then y is B_1 ,

also R₂: if x is A_2 then y is B_2 ,

...

also R_n: if x is A_n then y is B_n .

where x is an input fuzzy variable, y is an output fuzzy variable. A_i denotes the fuzzy sets of the *antecedent* fuzzy variable x and B_i denotes the fuzzy sets of the *consequent* fuzzy variable y . The membership functions of the fuzzy sets A_i and B_i are $\mu_{A_i}(x)$ and $\mu_{B_i}(y)$. If we assume the input is fuzzy singleton x_0 and the consequence is B^* , by using Mamdani's minimum operation rule R_c shown in (2.2.4.3) as our fuzzy implication function, the i th rule leads to the control decision

$$\mu_{B_i}(y) = \hat{\alpha}_i \wedge \mu_{B_i}(y) = \mu_{A_i}(x_0) \wedge \mu_{B_i}(y) \quad (2.3.4.1)$$

which implies that the membership function $\mu_{B_i}(y)$ of the inferred consequence B^* is given by

$$\mu_{B^*}(y) = \bigvee_{i=1}^n \mu_{B_i}(y) = \bigvee_{i=1}^n (\hat{\alpha}_i \wedge \mu_{B_i}(y)) \quad (2.3.4.2)$$

Defuzzification yields a nonfuzzy control action y_0 from an inferred fuzzy control action by finding the center of the area of $\mu_{B_1}(y)$. The calculation is done as follows [23]:

$$y_0 = \frac{\int y \mu_{B_1}(y) dy}{\int \mu_{B_1}(y) dy} \quad (2.3.4.3)$$

Chapter 3 Adaptive Leaky Bucket and Fuzzy Leaky Bucket

3.1 Adaptive Leaky Bucket

In an adaptive LB control approach (the interested reader may consult [37]), the token generation rate may be a function of the network load or congestion. In most cases the congestion of a network can be quantified by the switch buffer occupancy percentage of the network. We define such a measure as the *network occupancy ratio* if we assume lower threshold of network occupancy ratio is θ_1 and upper threshold of network occupancy ratio is θ_2 . We further assume a system evolving in discrete time. If at time slot k the token generation interval is y_k , then at time slot $(k+1)$ the token generation interval y_{k+1} can be calculated as:

If (network occupancy ratio at time slot $k < \text{threshold } \theta_1$),

then $y_{k+1} = \alpha_1 y_k$ ($\alpha_1 < 1$);

else

if (network occupancy ratio at time slot $k > \text{threshold } \theta_2$),

then $y_{k+1} = \alpha_2 y_k$ ($\alpha_2 > 1$);

else

if (threshold $\theta_1 < \text{network occupancy ratio at time slot } k < \text{threshold } \theta_2$),

then $y_{k+1} = y_k$.

where the thresholds θ_1 and θ_2 may be selected suitably so that certain performance measures are optimized. Furthermore, in this algorithm, α_1 (α_2) is the rate of the decrease

(increase) token generation interval. We will refer to them as the *token generation adaptation parameters*.

In this control framework, a control rule that depends on a number of thresholds (usually two) for LB parameter adjustment is employed. These thresholds represent high or low network occupancy or congestion.

Congestion in the network is detected when the network occupancy ratio exceeds the upper threshold θ_2 . In this case, the leaky bucket limits the transmission of traffic by changing the token generation interval γ (in this case the token generation interval is increased).

Lower network traffic load is also detected when the network occupancy ratio is less than the lower threshold θ_1 . In this case, the token generation interval is reduced.

The aim of adaptive control is to avoid the two extremes that may happen in basic leaky bucket scheme. In one extreme where the leaky bucket does not throttle the sources, most of the cell loss takes place inside the network, which affects the performance of other sources as well. In the other extreme, where the leaky bucket is “tightly” controlling the source, most of the cells are buffered in the leaky bucket queues, cell loss may occur at the access node even if the channel bandwidth to which the sources is connected is underutilized.

The challenge of this control scheme is to choose appropriate control parameters, which in our study are the network buffer thresholds.

3.2 Fuzzy Leaky Bucket Mechanism

In this section, we propose a modification of the adaptive leaky bucket based on fuzzy logic principles. We will refer to this system as the “fuzzy” leaky bucket. A fuzzy

LB (Figure 3-1) bears certain similarities with the adaptive LB and consists of a finite token buffer, a user buffer, a stream of arriving bursty cells and a fuzzy logic controller.

Figure 3-1 Fuzzy leaky bucket

In the fuzzy LB mechanism that we propose in this study the lower and higher network traffic loads do not correspond to *fixed* thresholds θ_1 and θ_2 . In order to develop a more realistic view of buffer occupancy the traditional notion of “buffer occupancy” shown in Figure 3-2, is appropriately modified with three fuzzy states as “SMALL, MEDIUM and LARGE”. This approach provides continuous set membership functions and implies a soft, gradual transition between small and medium as well as medium and large network loads. We utilize a triangular or trapezoidal symmetric shaped membership functions shown in Figure 3-3. The token generation interval y is determined by means of the following three rules:

Rule 1: If the network occupancy (x) is *small* (A_1), then the token generation interval y is *short* (B_1),

Rule 2: If the network occupancy (x) is *medium* (A_2), then the token generation interval y is *medium* (B_2),

Rule 3: If the network occupancy (x) is *large* (A_3), then the token generation interval y is *long* (B_3),

Figure 3-2 Partitioning of the buffer occupancy corresponding to two thresholds

Figure 3-3 Modified buffer occupancy corresponding to three fuzzy states

where x is an antecedent (input) fuzzy variable representing the network occupancy and y is an (consequent) output fuzzy variable for the token generation interval. Furthermore, A_i denotes the fuzzy sets of the antecedent fuzzy variable x (small, medium and large) and B_i denotes the fuzzy set of the consequent fuzzy variable y (short, medium and long) for $i = 1, 2, 3$. The membership functions of the fuzzy sets A_i and B_i are $\mu_{A_i}(x)$ and $\mu_{B_i}(y)$ as depicted in Figure 3-4.

Figure 3-4 Fuzzy variable membership functions

The calculation of the fuzzy LB follows the equation (2.3.4.1) - (2.3.4.3) presented in chapter 2. The graphic presentation of the calculation is shown in Figure 3-5. If the input is assumed x_0 , the assumed fuzzy membership functions are shown in Figure 3-4. Finally, a nonfuzzy control result y_0 can be obtained by finding the center of the area of $\mu_{B_i}(y)$. This can be achieved by using equation (2.3.4.3). In the discrete simulation [23], The computation of the center of the area can be approximately equal to

$$y_0 = \frac{\sum_{j=1}^n \mu_B(y_j) \cdot y_j}{\sum_{j=1}^n \mu_B(y_j)} \quad (3.2.1)$$

where n is the number of quantization levels of the output.

Figure 3-5 Fuzzy reasoning and implication of fuzzy LB

We readily see that the fuzzy LB approach modifies the traditional LB in three fields.

- 1) The fuzzy LB uses of fuzzy variables in place of numerical variables.
- 2) The fuzzy LB characterize the relations between input and output variables by fuzzy rules.
- 3) The fuzzy LB solves the complex relations by a fuzzy implication.

Chapter 4 Network Performance Evaluation

4.1 Introduction

In this chapter, the performance of the ATM network with the proposed fuzzy LB congestion control mechanism is investigated. In order to evaluate the effectiveness of the fuzzy LB, the performance of the ATM network with the adaptive LB mechanism is also studied and compared with the corresponding one of the fuzzy LB. Network parameters which significantly affect the performance are identified and optimized for the two control schemes. A comparison of the two control schemes is carried out under the same network condition and optimal parameters. Finally, the performance of the fuzzy LB and adaptive LB are evaluated under self-similar traffic load.

The performance analysis in this section is based on simulation combined with numerical optimization methods.

4.2 Network Simulation

4.2.1 Network Topology

Figure 4-1 illustrates a diagram of the simplified ATM network topology that will be studied. In this model, a “two hop” (one switch) network topology is assumed and the trunk line speed is set at 50 Mb/sec. Two model configurations are compared namely adaptive LB control and fuzzy LB control employed for policing the data sources.

Figure 4-1 ATM network topology

4.2.2 Quality of Service Parameters

The network performance is evaluated by two quality of service parameters: cell loss ratio and queuing delay.

Cell in an ATM network may be lost due to limited buffer capacity at switching/multiplexing nodes. In such case cells are lost from overflowing buffers fed by burst traffic during peak-rate emission phases. The effect of the burstiness in an ATM network can be limited by source policing using LB control. In our study, cell loss may happen in the user queue (LB queue) and the network switch queue (see Figure 4-2 network queuing model).

An overall cell loss ratio L is calculated within a relatively long simulation length interval (steady state condition) and defined as:

$$L = (L_n + L_u) / L_t \quad (4.2.1)$$

where L_n and L_u are, respectively, the number of data cell lost in network queue and user queue and L_t is the number of the total data cell that sources sent to the network during a replication of a simulation.

End to end delay on the other hand is a combination of queueing delay, propagation delay, convergence/segmentation delay, and switch delay. Convergence/segmentation delay is incurred when traffic is translated and segmented into cells by convergence/segmentation services at the sender, when the opposite operations are performed at the receiver. Convergence/segmentation delay and switch delay are normally invariant with respect to traffic conditions, and they may be ignored. We primarily focus in the variable queueing delay D defined as:

$$D = D_n + D_u \quad (4.2.2)$$

where D_n and D_u are, respectively, the network queue delay for data traffic and the user queueing delay for data traffic.

4.3 Simulation Model and Assumptions

4.3.1 Simulation Model

A detailed view of our simulation model containing a number of data and video sources is shown in Figure 4-2. Video is used as background traffic in all simulation experiments and it carries 25% of the total load. Furthermore, we assume that it is not controlled by a LB since video traffic is delay sensitive. Each data source is however policed by a LB. Both video inputs and the LB output are delivered to an ATM queue (representing a switch or multiplexer) which is assumed to have a 50 Mbit/sec service link.

Two simulation case studies are implemented using the SIMSCRIPT II.5 simulation language. In the first we employ an adaptive LB control scheme as described in section 3.1 while in the latter we develop a fuzzy LB control methodology outlined in section 3.2. The

simulation is done in discrete time and the simulation time unit corresponds to 0.00848 ms which is the service time of an ATM cell corresponding to a bit rate of 50Mbit/sec.

Figure 4-2 ATM simulation queueing model

4.3.2 Poisson Process Traffic Model

A Poisson batch arrival process is used to model the data traffic. The probability distribution of the batch length in each arrival follows a truncated geometric distribution.

This traffic model is consistent with results reported in [14]. We assume that the mean batch length of a data traffic is 20 cells, which is suggested for token ring traffic in [30]. Table 4-1 lists the average arrival rate of a data source in the different network data traffic load.

Table 4-1 Network traffic load vs. average Poisson batch arrival rate

Network Data Traffic Load ρ	Number of Data Sources	Average Data Arrival Rate(Batch/s)	Mean Data Rate(Mb/s)
40%	4	590	5
45%	4	663	5.625
50%	4	737	6.25
55%	4	811	6.875
60%	4	885	7.5

4.3.3 Self-Similar Traffic Model

Statistical analyses of large sets of traffic measurements [36] [16] from working packet networks show that, from a statistical view point, traditional modeling assumptions such as Poisson batch arrivals, exponential service and Markovian structure cannot sufficiently represent real traffic patterns. Since all these models have in common an asymptotically exponential decay of autocorrelation function and a rapidly decaying marginal distribution tail. Furthermore they lack a systematic way of simultaneously fitting both the empirical marginal distribution and the autocorrelation function. Instead, actual network traffic can be more accurately matched by self-similar (fractal) models.

We now recall some of the results in [36] about a set of ethernet LAN traffic measurements at the packet level. The observation was that the arrival rates measured over larger time scales (hours, minutes) are indistinguishable from those measured over smaller time scales (seconds, milliseconds). In particular, no natural length of a “burst” is discernible: at every time scale ranging from milliseconds to minutes and hours, bursts consist of

bursty subperiods separated by less bursty subperiods. This is the so called the “self-similar” property of ethernet traffic.

The crucial feature of self-similar processes is that they exhibit long range dependence (LRD), that is, their autocorrelation function $r(k)$ decays less than exponentially fast, has the form $r(k) \sim k^{-\beta}$, where $0 < \beta < 1$. The quantity $H = 1 - \beta/2$ is call the hurst parameter. This is in contrast to traditional stochastic models, all of which exhibit short range dependence. LRD processes are non-Markovian in nature and give rise to features and network behavior that are dramatically different from those of traditional short-range dependent processes. In particular, the latter give rise to a summable autocorrelation function $0 < \sum_k r(k) < \infty$, while long-range dependence implies non-summability of the correlation $\sum_k r(k) \rightarrow \infty$. The serious implication for ATM network design is that conclusions based on traditional models may not be applicable under self-similar traffic models. Therefore, in this thesis, the network performance will be also evaluated by using a self-similar traffic model.

There are some stochastic models which exhibit the self-similar property. Two of them, namely the exactly self-similar fractional Gaussian noise (FGN) and the asymptotically self-similar fractional autoregressive integrated moving-average (F-ARIMA) process, are the most commonly used [16].

The generation of long synthetic traces from self-similar Gaussian processes is performed by following the procedure in [16]. A Hosking's procedure is selected to generate a trace from self-similar Gaussian processes. For such a process X with mean $m = 0$, the conditional mean and variance X_k , given the past values $x_{k-1}, x_{k-2}, \dots, x_0$, may be written as:

$$m_k = E(X_k | x_{k-1}, x_{k-2}, \dots, x_0) = \sum_{j=1}^k \varphi_{kj} x_{k-j} \quad (4.3.1)$$

$$v_k = \text{var}(X_k | x_{k-1}, x_{k-2}, \dots, x_0) = \sigma^2 \left(\prod_{j=1}^k (1 - \varphi_{jj}^2) \right) \quad (4.3.2)$$

Here ϕ_{jj} is the j th partial correlation coefficient of $\{X_k\}$ and the ϕ_{kj} are partial linear regression coefficients. For simulating a sample process with correlation $r(k)$, we have the following algorithm:

1. Generate a starting value x_0 from a Gaussian distribution $N(0, v_0)$. Set $N_0 = 0, D_0 = 1$.
2. For $k = 1, \dots, n-1$, calculate $\phi_{kj}, j = 1, \dots, k$, recursively via the equations

$$N_k = r(k) - \sum_{j=1}^{k-1} \phi_{k-1,j} r(k) \quad (4.3.3)$$

$$D_k = D_{k-1} - N_k^2 / D_{k-1} \quad (4.3.4)$$

$$\phi_{kk} = N_k / D_k \quad (4.3.5)$$

$$\phi_{kj} = \phi_{k-1,j} - \phi_{kk} \phi_{k-1,k-j} \quad j = 1, \dots, (k-1) \quad (4.3.6)$$

calculate $m_k = \sum_{j=1}^k \phi_{kj} x_{k-j}$ and $v_k = (1 - \phi_{kk}^2) v_{k-1}$. Generate x_k from the Gaussian distribution $N(m_k, v_k)$.

This method is applicable to any causal Gaussian process as long as the correlation function $r(k)$ is known.

The generation of a self-similar process Y_k with a desirable marginal distribution can be done by the following transformation [16]:

$$Y_k = h(X_k) = F_Y^{-1}(F_X(X_k)) \quad k = 1, 2, \dots \quad (4.3.7)$$

x_k can be viewed as a Gaussian process with zero mean, unity variance and autocorrelation function $r(k)$. $F(x)$ is its marginal cumulative probability function. $F_Y(y)$ is a marginal cumulative probability function of process Y_k .

In this thesis, $F_Y(y)$ is obtained by inverting the empirical distribution shown in Figure 4-3. We assume that the self-similar Gaussian process X has the following Hurst parameter $H = 0.6$, $H = 0.75$, and $H = 0.95$. Three traces with same Poisson marginal distribution and different autocorrelation functions ($H = 0.6$, $H = 0.75$, and $H = 0.95$) are generated as our data sources in the simulation. The mean data rate for each data source was measured to be 1.5Mbit/s.

Figure 4-3 Video frame empirical distribution

4.3.4 Video Trace

The video source traces that were selected to drive the simulations are segments from the movie “Star Wars” that were collected from Internet at the Bellcore site ([ftp.bellcore.com](ftp://ftp.bellcore.com)). Figure 4-4 shows a plot of the bit rate for a segment of the “Star Wars”. The segment is approximately two minutes long. Each video frame was appropriately parsed into a variable number of ATM cells depending on the frame size and assuming an AAL5 connection. The mean bit rate was measured to be 6.25Mbit/sec for each source. The aggregate mean bit rate of video sources in the simulation is 12.5 Mbit/sec corresponding to two video sources. The video cells are entering the network buffer unrestricted along with the output traffic of each LB.

Figure 4-4 Bit rate of video traffic

4.3.5 Transient Time and Variance Reduction

In Figure 4-5, an estimated mean queueing delay versus simulation run time is plotted. The mean queueing delay for data cells was computed at 0.2 second intervals. Steady state conditions are reached after 3 second according to Figure 4-5. During the first 1.2 second, the downward bias is observed due to the empty buffer initial condition. In all simulations data collection starts after 3 seconds of warm-up period to avoid the initial transient time.

Figure 4-5 Estimated mean queueing delay vs. simulation run time

Data variance reduction is achieved by simulating different replications. The number of replications is determined by the following procedure.

The half length (h.l) of a $100(1-\alpha)\%$ confidence interval for a mean θ , based on the t distribution, is given by [1] [21]

$$h.l = t_{\alpha/2, R-1} \hat{\sigma}(\hat{\theta}) \quad (4.3.8)$$

where α is the level of significance, $\hat{\sigma}(\hat{\theta}) = S/(\sqrt{R})$. S denotes sample standard deviation, while R is the number of replications.

To meet a mean queueing delay half length criterion of $\varepsilon = 0.4ms$ with confidence 95% ($\alpha = 5\%$) R must be chosen so that

$$h.l = \frac{t_{\alpha/2, R-1} S}{\sqrt{R}} \leq \varepsilon \quad (4.3.9)$$

which provides an estimation of the number of replication as follows:

$$R \geq \left(\frac{t_{\alpha/2, R-1} S}{\varepsilon} \right)^2 \quad (4.3.10)$$

4.3.6 Assumptions for the Network Parameters

Table 4-2 summarizes the assumptions of network parameters that are used in the simulations for both adaptive LB and fuzzy LB. The token buffer size of LB is determined by the average burst length in the token ring network[30].

Table 4-2 Network parameter assumptions

Description	Assumption of adaptive LB	Assumption of fuzzy LB
Simulation Length for Each Replication	18 sec	18 sec
Number of Replications	20	20
Network Buffer Size	150 cells	150 cells
User Buffer Size	200 cells	200 cells
Number of Video Sources	2	2
Average Arrival rate of a Video Source	6.25 Mb/s	6.25 Mb/s
LB Token Buffer Size	20 cells	20 cells
Initial Token Generation Interval	equal to average cell interarrival time	equal to average cell interarrival time
α_1 : Decrease Rate of Token Generation Interval	0.85	N/A
α_2 : Increase Rate of Token Generation Interval	2	N/A
ATM Network Switch Service Time (or time slot)	0.00848ms	0.00848ms

4.4 Preliminary Results and Methodology for Adaptive LB

4.4.1 Performance Measures

In our simulation study, for the comparison of the effectiveness of the fuzzy and non-fuzzy LB control schemes we developed the following performance methodology which relies on a constrained optimization problem that involves the total cell delay D and loss probability L as follows:

Minimize (with respect to x) the delay $D(x)$ so that

$$L(\underline{x}) \leq L_p \quad (4.4.1)$$

where L_p is a given number that can be viewed as the maximum acceptable cell loss probability. Furthermore, the vector $\underline{x}$ can be viewed as a control parameter (e.g., buffer occupancy thresholds).

According to the Kuhn - Tucker conditions [10] [25], if the vector $\underline{x}^*$ be an optimal point for the problem above then there is a number $\lambda \geq 0$ such that

$$\nabla_x D(\underline{x}^*) + \lambda \nabla_x (L(\underline{x}^*) - L_p) = 0 \quad (4.4.2)$$

$$\lambda (L(\underline{x}^*) - L_p) = 0 \quad (4.4.3)$$

Equation (4.4.2) yields.

$$\nabla_x (D(\underline{x}^*) + \lambda (L(\underline{x}^*) - L_p)) = 0 \quad (4.4.4)$$

which is true if the Lagrangian cost function defined by

$$C(\underline{x}, \lambda) = D(\underline{x}) + \lambda (L(\underline{x}) - L_p) \quad (4.4.5)$$

attains a minimum at $\underline{x}^*$.

Therefore in our simulation experiments, in order to determine the solution $\underline{x}^*$ for the optimization problem for each λ we minimize the Lagrangian cost function $C(\underline{x}, \lambda)$ with respect to $\underline{x}$ and we check the following optimality condition

$$(L(\underline{x}) - L_p) \approx 0 \quad (4.4.6)$$

If the condition is not true the value of λ is adjusted appropriately until the condition (4.4.6) is satisfied, in which case the Lagrangian cost minimizer $\underline{x}^*$ is selected as the solution of the optimization problem (4.4.1). For the sake of completeness we assume that both $D(\cdot)$ and $L(\cdot)$ are convex functions of $\underline{x}$ which is the case in most queueing problems and evidenced by a number of plots (see Figure 4-6 and Figure 4-7).

Figure 4-6 Cell loss $L(\theta_1, \theta_2)$ using the Poisson process for data sources ($\rho = 80\%$)

Figure 4-7 Total mean queueing delay $D(\theta_1, \theta_2)$ using the Poisson process for data sources ($\rho = 65\%$)

4.4.2 Cell Loss Criteria L_p

In order to determine a reasonable upper bound L_p for the cell loss for each control parameter set (θ_1, θ_2) we first find the minimum value of $L(\theta_1, \theta_2)$. The minimization was performed by using a two-dimensional mean fielding annealing (MFA) optimization algorithm (the interested reader may check [29] [6] references listed therein) which was appropriately incorporated into our simulation model. Initially our system was fed by video and

Poisson batch data traffic sources. The data traffic is policed (controlled) by an adaptive LB. $L(\theta_1, \theta_2)$ is evaluated at discrete steps over the two parameter (θ_1, θ_2) space by the MFA algorithm. The range of values of the parameter set (θ_1, θ_2) for the optimization process is shown in Table 4-3.

Table 4-3 MFA range of values for (θ_1, θ_2)

Factors	Minimum Value	Maximum Value	Stepsize
θ_1	0.22	0.55	0.03
θ_2	0.55	0.88	0.03

A phase plot of the parameter set (θ_1, θ_2) during a MFA optimization is shown in Figure 4-7. In this particular case, the network traffic load was maintained at 70%, including 25% background traffic. Figure 4-8 indicates the point which is found as the minimizer of $L(\dots)$ by the MFA algorithm. Similar experiments were performed for various traffic loads. The minimum cell loss ratio for each case is shown in the second column of Table 4-4. Simulations were also conducted for determining $L(\dots)$ at the extreme points of the parameter set (θ_1, θ_2) and finally the value of L_p was selected as approximately twice the value of the minimum cell loss ratio determined previously. The values of L_p for various network traffic load conditions are shown in the third column of Table 4-4. Such a choice of L_p guarantees a feasible solution for the delay optimization problem (4.4.1) that we described in the previous section and we will consider in greater detail in the rest of our study.

Table 4-4 Minimum cell loss ratio in adaptive LB

Network traffic load	Minimum cell loss	L_p
65%	0.0009	0.002
70%	0.0018	0.004
75%	0.0031	0.006
80%	0.0059	0.012
85%	0.0290	0.060

Figure 4-8 (θ_1, θ_2) trace recorded by running MFA

4.5 Simulation Results of Adaptive LB

4.5.1 Design Optimization for Adaptive LB

Optimization of the network performance is achieved by solving the optimization problem (4.4.1) with respect to the control parameters θ_1, θ_2 . In this study, an analytic solution of the appropriate cost functions is not available therefore statistical estimates were obtained from simulation experiments.

According to equation (4.4.5), the Lagrangian cost function for adaptive LB in this case is defined by:

$$C(\theta_1, \theta_2) = D(\theta_1, \theta_2) + \lambda (L(\theta_1, \theta_2) - L_p) \quad (4.5.1)$$

with the following condition of optimality

$$(L(\theta_1, \theta_2) - L_p) \approx 0 \quad (4.5.2)$$

4.5.2 Adaptive LB Simulation Results Using Poisson Batch Arrival Process for Data Sources

In our first experiment we determine the optimal parameters (θ_1, θ_2) under each network traffic load using Poisson batch arrival process for data sources. In order to keep the control parameter space small we selected the adaptation parameters for the token generation time interval to be $\alpha_1=0.85$ and $\alpha_2=2$ [37].

For each positive λ a series of simulations is performed incorporating an MFA algorithm which determines the direction of descent for the Lagrangian cost function (4.5.1) in the parameter space (θ_1, θ_2) . When this numerical optimization is completed the optimality condition (4.5.2) is evaluated and if satisfied the experiment is concluded and the cell delay $D(\dots)$ is evaluated. In case the relation (4.5.2) is not satisfied the Lagrange multiplier λ is adjusted and a new numerical optimization is initiated.

The optimal values of (θ_1, θ_2) for various network loads are shown in the second column of Table 4-5. We readily observe that under a network traffic load of 85% the token generation interval is reduced by a factor of 0.85 when the network occupancy ratio is less than 0.34 and it is doubled for network occupancy ratio greater than 0.67. Similar results were obtained for other traffic loads as indicated in Table 4-5.

The queueing delay $D(\dots)$ is calculated as 95% confidence intervals using the optimal parameters for each traffic load, as shown in Figure 4-9. This figure will be used for comparison with similar results that will be derived by the fuzzy LB controller as outlined in the following section.

Table 4-5 Network traffic load vs. optimal parameters in adaptive control corresponding to Poisson batch arrival process for data sources

Network Traffic Load ρ	Optimal Parameters (θ_1, θ_2)	λ Value	Condition for Optimality
65%	(0.46, 0.79)	5500	$ L(\theta_1, \theta_2) - 0.002 = 0.0001$
70%	(0.43, 0.76)	6000	$ L(\theta_1, \theta_2) - 0.004 = 0.0001$
75%	(0.40, 0.73)	5000	$ L(\theta_1, \theta_2) - 0.006 = 0.0002$
80%	(0.37, 0.70)	1000	$ L(\theta_1, \theta_2) - 0.012 = 0.0007$
85%	(0.34, 0.67)	500	$ L(\theta_1, \theta_2) - 0.060 = 0.0020$

Figure 4-9 Queuing delay in adaptive control using Poisson data sources.

4.5.3 Adaptive LB Simulation Results Using Self-Similar Data Traffic

In the second experiment we follow similar steps as previously, however we incorporate a self similar process as the data arrival process.

Three different data traces with Hurst parameter 0.6, 0.75, and 0.95 were generated to drive the network simulation. The optimal parameters (θ_1 , θ_2) for the optimization problem are determined for the three traces with different Hurst parameters corresponding to data traffic.

For the cases of the Hurst parameter being 0.6 and 0.75 we observed that the optimal values for (θ_1 , θ_2) were close to the ones determined in the first experiment involving Poisson arrival processes. The results are summarized in the Table 4-6 and Table 4-7. Computations for the case of the Hurst parameter being equal to 0.95 provided different results that are summarized in Table 4-8.

Table 4-6 Network traffic load vs. optimal parameters in adaptive control corresponding to self-similar data sources with Hurst parameter = 0.6

Network Traffic Load ρ	Optimal Parameters(θ_1, θ_2)	λ Value	Condition for Optimality
65%	(0.46, 0.79)	5500	$ L(\theta_1, \theta_2) - 0.002 = 0.0002$
70%	(0.43, 0.76)	6000	$ L(\theta_1, \theta_2) - 0.004 = 0.0003$
75%	(0.40, 0.73)	5000	$ L(\theta_1, \theta_2) - 0.006 = 0.0005$
80%	(0.37, 0.70)	1000	$ L(\theta_1, \theta_2) - 0.012 = 0.0008$
85%	(0.34, 0.67)	500	$ L(\theta_1, \theta_2) - 0.060 = 0.0050$

Table 4-7 Network traffic load vs. optimal parameters in adaptive control corresponding to self-similar data sources with Hurst parameter = 0.75

Network Traffic Load ρ	Optimal Parameters(θ_1, θ_2)	λ Value	Condition for Optimality
65%	(0.46, 0.79)	5500	$ L(\theta_1, \theta_2) - 0.002 = 0.0002$
70%	(0.43, 0.76)	6000	$ L(\theta_1, \theta_2) - 0.004 = 0.0003$
75%	(0.40, 0.73)	5000	$ L(\theta_1, \theta_2) - 0.006 = 0.0004$
80%	(0.37, 0.70)	1000	$ L(\theta_1, \theta_2) - 0.012 = 0.0009$
85%	(0.34, 0.67)	500	$ L(\theta_1, \theta_2) - 0.060 = 0.0010$

Table 4-8 Network traffic load vs. optimal parameters in adaptive control corresponding to self-similar data sources with Hurst parameter = 0.95

Network Traffic Load ρ	Optimal Parameters(θ_1, θ_2)	λ Value	Condition for Optimality
65%	(0.46, 0.79)	5500	$ L(\theta_1, \theta_2) - 0.002 = 0.0002$
70%	(0.43, 0.73)	5500	$ L(\theta_1, \theta_2) - 0.004 = 0.0004$
75%	(0.40, 0.70)	5000	$ L(\theta_1, \theta_2) - 0.006 = 0.0006$
80%	(0.37, 0.67)	1000	$ L(\theta_1, \theta_2) - 0.012 = 0.0010$
85%	(0.34, 0.64)	500	$ L(\theta_1, \theta_2) - 0.060 = 0.0030$

The queueing delays are calculated as 95% confidence intervals using optimal parameters, as shown in Figure 4-10. Figure 4-10 shows that the mean queueing delay is increased when H increases. Under high bursty traffic scenario ($H=0.95$), the queueing delay is increased dramatically. This figure will also be used for comparison with similar results that will be derived by the fuzzy LB controller in the following section.

Figure 4-10 Queueing delay in adaptive control using self-similar data sources with different hurst parameters.

4.6 Fuzzy LB Controller

4.6.1 Implementation of the Fuzzy LB Control

In this section we present the development of a fuzzy LB controller based on the knowledge gained in the experiments that we have already performed for Adaptive LB. The goal of this study is to provide a comparison of the performance of the two control

schemes. The design procedure includes the following steps: first, we establish the fuzzy control rules outlined in Chapter 3. Second, we design and tune the parameters of fuzzy sets (or membership functions) that characterize the controller. The rules we formulated are based on a three level network buffer occupancy and follow the design of the adaptive LB described in the previous sections. In our framework the network buffer occupancy may be characterized as *small*, *medium* or *large*. As we have also summarized in section 3.2, the token generation time interval can also be characterized as *short*, *medium* or *long*, respectively. In this section, we concentrate on designing and adjusting the corresponding membership functions of the associated fuzzy sets.

In order to present our methodology we focus on the following example. We consider a scenario of the network traffic load being 80%. In this case, the optimal parameters θ_1 and θ_2 for adaptive LB were determined to be 0.37 and 0.70, respectively. Since the network buffer size is assumed to be 150 cells, the absolute threshold $\bar{\theta}_1$ for low network occupancy is 56 cells ($150 \times 0.37 = 56$) and similarly the absolute threshold $\bar{\theta}_2$ for high occupancy is 105 cells ($150 \times 0.70 = 105$). The three states of the network buffer represented by the absolute thresholds $\bar{\theta}_1$ and $\bar{\theta}_2$ are shown in Figure 4-11 (we use dashed variables to denote absolute quantities).

The fuzzy membership functions for *small* and *large* are now obtained by modifying Figure 4-11 in the following intuitive manner. Let the line segment AB centered around threshold $\bar{\theta}_1$ have a negative slope $-k_2$ and let the line CD centered around threshold $\bar{\theta}_2$ have a positive slope k_2 . The trapezoidal shaped membership functions for *small* and *large* are obtained as shown in Figure 4-12. The line segments AB and CD can be turned around the centre fixed point U (or V). The fuzzy membership function of *medium* is assigned a triangular shape as shown in Figure 4-12. The fixed point F is in the middle of $\bar{\theta}_1$ and $\bar{\theta}_2$ corresponding to a value of 80 as shown in Figure 4-12.

Figure 4-11 Network buffer occupancy as determined by optimal parameters.

Figure 4-12 Modified network buffer occupancy.

The line segment FE has a positive slope k_1 and the line segment FG has a negative slope $-k_1$. Points E and G can be modified however point F is assumed fixed. In this fashion k_1 and k_2 can be specified as the two variable control parameters that can be used for

the optimization of the fuzzy LB performance. The range of values selected for k_1 and k_2 for our experiment is shown in Table 4-9.

Table 4-9 range of values for the analysis of the fuzzy LB control

Factor	Point1	Point2	Point3	Point4	Point5	Point6	Point7	Point8	Point9	Point10
k_1	1/44	1/42	1/40	1/38	1/36	1/34	1/32	1/30	1/28	1/26
k_2	1/48	1/44	1/40	1/36	1/32	1/28	1/24	1/20	1/16	1/12

We now present the design procedure of the consequent membership functions (see Figure 4-13). The portion of the membership function corresponding to *medium* congestion is assigned a symmetrical triangular shape as shown in Figure 4-13. In particular point x_I corresponding to a token generation time interval equal to T_m which is the mean data traffic interarrival time for each case. Furthermore, based on the design of the adaptive leaky bucket controller we select $x_k = \alpha_2 T_m$. Since $\alpha_2 = 2$, then $x_k = 2T_m$. The portion of the function for the *short* token generation time interval has a trapezoidal shape with the outer left edges located at point $(x_L, 1) = (0.85T_m, 1)$ and $(x_L, 0)$. Finally, the portion of the membership function for the *long* token generation time interval has trapezoidal structure with the two rightmost edges located at $(x_I, 0)$ and $(x_k, 1)$.

Figure 4-13 Membership function of token generation interval

4.6.2 Cell Loss Characterization by Applying Fuzzy LB

Cell loss characterization by using fuzzy LB is performed in a similar manner as in section 4.4.2. The modification is that instead of parameters θ_1 and θ_2 two parameters (k_1 , k_2) are selected for analysis the cell loss $L(\dots)$ in fuzzy LB. The minimizer of $L(\dots)$ for various traffic load is evaluated at discrete intervals over the two parameter (k_1 , k_2) space by MFA algorithm. The minimum cell loss ratio for each case is shown in Table 4-10.

By comparing the results listed in Table 4-4 and Table 4-10 we can see that the fuzzy LB scheme yields better minimum cell loss ratio than an adaptive LB does. Therefore the L_p defined in third column of Table 4-4 can be also used for fuzzy LB simulation. Furthermore, it is fair to compare the two control schemes by using the same cell loss ratio criteria L_p .

Table 4-10 Minimum cell loss ratio in fuzzy LB

Network traffic load	Minimum cell loss
65%	0.0001
70%	0.0009
75%	0.0015
80%	0.0037
85%	0.0089

4.7 Fuzzy LB Simulation Results

4.7.1 Fuzzy LB Simulation Results Using Poisson Process Data Traffic

We now perform an optimization for the determination of the optimal fuzzy LB control parameters. The modified Lagrangian cost function for the fuzzy LB is defined with

respect to the control parameter set (k_1, k_2) by $C(k_1, k_2)$ and the corresponding cell loss ratio by $L(k_1, k_2)$. Similarly as before the optimal values for the parameters (k_1, k_2) , when the system is driven by Poisson batch arrival data traffic, are shown in Table 4-11. The queueing delay is calculated as 95% confidence intervals using optimal parameters for each traffic load, as shown in Figure 4-14.

Table 4-11 Network traffic load vs. optimal parameter in fuzzy control corresponding to Poisson data sources

Network Traffic Load ρ	Optimal Parameters (k_1, k_2)	λ Value	Condition for Optimality
65%	(1/30, 1/24)	5500	$ L(k_1, k_2) - 0.002 = 0.00008$
70%	(1/32, 1/28)	6000	$ L(k_1, k_2) - 0.004 = 0.00007$
75%	(1/34, 1/32)	5000	$ L(k_1, k_2) - 0.006 = 0.00009$
80%	(1/36, 1/36)	1000	$ L(k_1, k_2) - 0.012 = 0.0006$
85%	(1/38, 1/40)	500	$ L(k_1, k_2) - 0.060 = 0.0010$

Figure 4-14 Queueing delay in fuzzy control using Poisson data sources.

4.7.2 Fuzzy LB Simulation Results Using Self-Similar Data Traffic

For the cases of the self-similar process data arrival with Hurst parameter being 0.6 or 0.75 we observed that the optimal values for (k_1, k_2) were close to the ones determined in the previous experiment involving Poisson arrival processes. The results are summarized in Table 4-12, Table 4-13 and Table 4-14.

Table 4-12 Network traffic load vs. optimal parameters in fuzzy control corresponding to self-similar data sources with Hurst parameter = 0.6

Network Traffic Load ρ	Optimal Parameters (k_1, k_2)	λ Value	Condition for Optimality
65%	(1/30, 1/24)	5500	$ L(k_1, k_2) - 0.002 = 0.0001$
70%	(1/32, 1/28)	6000	$ L(k_1, k_2) - 0.004 = 0.0002$
75%	(1/34, 1/32)	5000	$ L(k_1, k_2) - 0.006 = 0.0003$
80%	(1/36, 1/36)	1000	$ L(k_1, k_2) - 0.012 = 0.0009$
85%	(1/38, 1/40)	500	$ L(k_1, k_2) - 0.060 = 0.0010$

Table 4-13 Network traffic load vs. optimal parameters in fuzzy control corresponding to self-similar data sources with Hurst parameter = 0.75

Network Traffic Load ρ	Optimal Parameters (k_1, k_2)	λ Value	Condition for Optimality
65%	(1/30, 1/24)	5500	$ L(k_1, k_2) - 0.002 = 0.0001$
70%	(1/32, 1/28)	6000	$ L(k_1, k_2) - 0.004 = 0.0001$
75%	(1/34, 1/32)	5000	$ L(k_1, k_2) - 0.006 = 0.0002$
80%	(1/36, 1/36)	1000	$ L(k_1, k_2) - 0.012 = 0.0009$
85%	(1/38, 1/40)	500	$ L(k_1, k_2) - 0.060 = 0.0030$

Table 4-14 Network traffic load vs. optimal parameters in fuzzy control corresponding to self-similar data sources with Hurst parameter = 0.95

Network Traffic Load ρ	Optimal Parameters (k_1, k_2)	λ Value	Condition for Optimality
65%	(1/32, 1/28)	5500	$ L(k_1, k_2) - 0.002 = 0.0002$
70%	(1/34, 1/32)	5500	$ L(k_1, k_2) - 0.004 = 0.0003$
75%	(1/36, 1/34)	5000	$ L(k_1, k_2) - 0.006 = 0.0004$
80%	(1/36, 1/36)	1000	$ L(k_1, k_2) - 0.012 = 0.0010$
85%	(1/38, 1/40)	500	$ L(k_1, k_2) - 0.060 = 0.0020$

The queueing delays are calculated as 95% confidence intervals using optimal parameters, as shown in Figure 4-15. For the cases of Hurst parameter being 0.6 and 0.75 the mean queueing delays are similar. However, a dramatic delay increase is observed for the case of Hurst parameter being 0.95.

Figure 4-15 Queueing delay in fuzzy control using self-similar data sources with different hurst parameter

4.8 Comparison of the Two LB Control Schemes

In this section, we present a comparison of the relative performance of the adaptive LB versus the fuzzy LB controller. First, we compare the total mean queuing delay that was derived by Poisson process for data sources as shown in Figure 4-16. Second, we compare the results that were derived by self-similar data sources as illustrated in Figure 4-17, Figure 4-18 and Figure 4-19.

By comparing the relative performance we can immediately observe that for the same blocking probability the total mean delay corresponding to the fuzzy control scheme is less than the one corresponding to the common adaptive LB control. This is evident in all cases involving different network loads and degrees of burstiness (Figure 4-16 - Figure 4-19).

It is interesting to observe that the system delays are comparable for all cases involving Poisson and self-similar data traces with Hurst parameters 0.6 and 0.75. Furthermore, we can readily observe a dramatic delay increase for the case Hurst parameter $H = 0.95$ due to the high burstiness of input traffic. This observation may also be contrasted with the fact that the optimal control parameters were found to be roughly the same for the first three cases (Poisson and self-similar data traces with Hurst parameters 0.6 and 0.75) but not for the fourth one (Hurst parameter $H = 0.95$).

Therefore, the fuzzy LB control scheme is more effective than adaptive LB control scheme. The research results also indicates that the fuzzy set theory can provide a robust mathematical frame work for dealing with real world imprecision and that the fuzzy approach exhibits a soft behavior, i.e., it has a greater ability to adapt to the rapid stochastic changes of the network traffic condition.

Figure 4-16 Comparison of the queueing delay (Poisson arrival process for data sources).

Figure 4-17 Comparison of the queueing delay (self-similar traffic $H=0.6$).

Figure 4-18 Comparison of the queuing delay (self-similar traffic $H=0.75$).

Figure 4-19 Comparison of the queuing delay (self-similar traffic $H=0.95$).

Chapter 5 Conclusions and Suggestions for Further Study

5.1 Summary and Conclusions

The main objective of this thesis is to present a new congestion control mechanism that introduces a leaky bucket control scheme based on fuzzy logic principles. We have achieved this goal by studying the non-fuzzy adaptive LB mechanism, developing a fuzzy LB control mechanism (by introducing a fuzzy rule and appropriate fuzzy variables) and comparing the two schemes.

The fuzzy LB proposed in this thesis is designed by formulating a set of control rules using fuzzy logic principles, identifying control parameters, and optimizing ‘soft’ bound membership functions. In such a framework, the parameters such as token generation interval can be tuned dynamically according to the network condition. Finally, intelligent reasoning was introduced to LB control scheme by employing a set of fuzzy rules.

In the adaptive LB control scheme, the decision making depends only on the two thresholds (“hard” bound). If the thresholds are not chosen properly, the control scheme may fail. In fuzzy LB control scheme, the decision making is a continuous function of network buffer occupancy (not two thresholds). Therefore, the “hard” bound effect in the adaptive LB has been eliminated in favor of soft fuzzy membership functions.

In order to evaluate the effectiveness of the fuzzy LB, network parameters which affect the performance have been identified and optimized for both adaptive and fuzzy LB control schemes. A modified constrained optimization problem that involves the total cell

delay D and loss probability L has been developed as follows (vector $\mathbf{x}$ is control parameters for each control scheme):

Minimize (with respect to $\mathbf{x}$) the delay $D(\mathbf{x})$ so that $L(\mathbf{x}) \leq L_p$

The problem was transformed to a Lagrangian optimization problem which was solved by using Mean Field Annealing (MFA), an efficient stochastic optimization technique.

Two simulation case studies were implemented using the SIMSCRIPT II.5 simulation language. The first employed the adaptive LB control scheme, while the second was based on the fuzzy LB control scheme.

The optimal parameters for each control scheme were determined by incorporating MFA with ATM network simulations. All network performance presented in this thesis were evaluated using optimal congestion control parameters.

Finally, the performance of the fuzzy LB and the adaptive LB were evaluated under Poisson process for data sources and self-similar traffic for data sources. A comparison of the two approaches was carried out under the same network condition and optimal parameters.

The improvement of fuzzy LB congestion control scheme was demonstrated by comparing the relative performance of the adaptive LB versus the fuzzy LB controller. For the same blocking probability, the total mean delay corresponding to the fuzzy control scheme is less than the one corresponding to the common adaptive LB control. This is evident in all cases involving different network loads and degrees of burstiness. We therefore conclude that the “soft” bound membership functions in fuzzy LB control scheme may lead to more robust and better decision making than the “hard” bound (threshold) one in the adaptive LB control scheme.

5.2 Suggestions for Further Investigations

5.2.1 Expansion of Control Parameter Space

In our study, in order to keep the simulation effort at a manageable level we used a reduced control parameter space for the optimizations. We therefore specified two control variables $((\theta_1, \theta_2)$ or $(k_1, k_2))$ while in a more general case other variables such as the token generation rate could also be identified as control parameters and used in the optimization. It still remains an open issue to choose additional control parameters in the fuzzy LB scheme.

We believe that our study will be a useful starting point that will motivate further research towards this direction. Such approaches may require elaborate optimization methodologies which can provide solution with reasonable computing effort in problem with large state spaces.

5.2.2 Video Rate Control

In this study traffic control was not applied to video sources since video traffic has strict transmission requirements in terms of delay and cell loss. It may be however possible to investigate a fuzzy control scheme that regulates video in a similar manner as we already did for data. Possible “cross - coupling” of the video and the data controller may also be considered. Such coupling can be (loosely) expressed in the following fuzzy rule:

If the change of video rate is (*small, medium, large*) and the network occupancy is (*small, medium, large*) then adjust token generation rate to be (*small, medium, large*).

Again the dimensionality of the problem is increased thus necessitating advanced optimization methodologies.

References

- [1] J. Banks and J.S. Carson, *Discrete event system simulation*, Prentice-Hall, 1984.
- [2] A. W. Berger, "Performance analysis of a rate control throttle where tokens and jobs queue," *IEEE J. Sel. Areas Commun.*, Vol.9, No. 3, pp. 325-334, Apr. 1991.
- [3] J.C. Bezdek, "Fuzzy models - what are they, and why?," *IEEE Trans. Fuzzy System*, Vol.1, No.1, pp. 1 - 6, February 1993.
- [4] A. R. Bonde and S. Ghosh, "A comparative study of fuzzy versus "fixed" thresholds for robust queue management in cell - switching networks," *IEEE/ACM Trans. Networking*, Vol. 2, No. 4, pp. 337 - 343, August 1994.
- [5] R. Cheng and C. Chang, "Design of a Fuzzy Controller for ATM Networks," *IEEE/ACM Transactions on Networking*, Vol.4 No. 3, pp.460 - 469, June 1996.
- [6] M. Devetsikiotis and K. Townsend, "Statistical optimization of dynamic importance sampling parameters for efficient simulation of communication networks," *IEEE/ACM. Trans. Net.*, Vol.1(3), pp.293 - 305, Jun. 1993.
- [7] D. Dubois, N. D. Georganas and E. Horlait, "A QoS selector for multimedia application on ATM networks," *Proc. ICC '94*, pp.160 - 164, 1994.
- [8] A. E. Eckberg, "B-ISDN/ATM traffic and congestion control," *IEEE Network*, pp.28-37, Sept.1992.
- [9] J. Favilla, "Fuzzy traffic control: adaptive strategies," *Second IEEE International Conference on Fuzzy Systems*, pp. 507 - 511, 1993.
- [10] R. Fletcher, "*Practical methods of optimization*," Vol.2, John Wiley & Sons, 1981.
- [11] G. Gallassi, G. Rigolio and L. Fratta, "ATM: bandwidth assignment and bandwidth enforce-

- ment policies," *GLOBECOM*, pp. 49.6.1-49.6.6, 1989.
- [12] S. J. Golestani, "Congestion-free communication in broadband packet network," *IEEE ICC*, pp. 308.3.1-308.3.6, 1990.
- [13] I. W. R. Handel and M. n. Huber, "*Integrated broadband networks*", Addison-Wesley, 1991.
- [14] H. Heffes and D. M. Lucantoni, "A Markov modulated characterization of packet voice and data traffic and related statistical multiplexer performance," *IEEE Jour. Sel. Areas in Commun.*, Vol. 4, No.6, 1986.
- [15] D. Hong and T. Suda, "Congestion control and prevention in ATM networks," *IEEE Network Magazine*, pp. 10-16, July 1991.
- [16] C. Huang, M. Devetsikiotis, I.E. Lambadaris and A. R. Kaye, "Modeling and Simulation of self-similar bit rate compressed video: a unified approach," *SIGCOMM '95*, pp.114-125, 1995.
- [17] D. Jensen, "B-ISDN network management by a fuzzy logic controller," *GLOBECOM '94*, pp. 799 - 804, 1994.
- [18] J. Jungok Bae and T. Suda, "Survey of traffic control schemes and potocols in ATM networks," *Proc. of the IEEE*, Vol.79, No. 2, pp. 170-189, Feb. 1991
- [19] B. Kosko, *Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence*. Englewood Cliffs, NJ: Prentice-Hall, 1992.
- [20] J. Kurose, "Open issues and challenges in providing quality of service guarantees in high-speed networks," *Computer Communication Review*, Vol. 23, No.1, pp. 6 - 15, Jan. 1993.
- [21] A. Law and D. Kalton, "*Simulation modeling and analysis*," New York, McGraw - Hill Inc,

1991.

- [22] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller - part I," *IEEE Trans. Systems, man, and cybernetics*, Vol. 20, No. 2, pp. 404 - 418, March 1990.
- [23] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller - part II," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. 20, No. 2, pp. 419 - 435, March 1990.
- [24] W. C. Lee, and M. G. Hluchyj, "Dynamic connection management for call-level QoS guarantee in integrated communication networks," *ICC'94*, pp. 1073 - 1082, 1994.
- [25] D. G. Luenberger, "*Linear and Nonlinear Programming*," second edition, 1984.
- [26] W. Matragi and K. Sohraby, "Combined Reactive/Preventive Approach for Congestion Control in ATM Networks," to appear.
- [27] M. Murata, Y. Qie, T. Suda, and H. Miyahara, "Analysis of a Discrete-time Single Server Queue with Bursty Inputs for Traffic Control in ATM Networks," *GLOBECOM'89*, pp. 49.5.1 - 49.5.7, 1989.
- [28] T. D. Ndousse, "Fuzzy neural control of voice cells in ATM networks," *IEEE JASC*, Vol. 12, No. 9, pp.1488 - 1494, 1994.
- [29] R. Nobakht, D. Van Den Bout, K. Townsend, and S. Ardalan, "Optimization of transmitter and receiver filters for digital communication system using mean field annealing," *IEEE Jour. Sel. Areas in Commun.*, Vol.8(8), pp.1472 - 1480, Oct. 1990.
- [30] R. O. Onvural, "Asynchronous transfer mode networks." 1st edition, 1994, Arkch House
- [31] E. P. Rathgeb, "Modeling and performance comparison of policing mechanisms for ATM networks," *IEEE J. Select. Areas Commun.*, Vol. 9, No. 3, pp. 325-334, April 1991.
- [32] M. Sidi, W. Z. Liu, I. Cidon, and I. Gopal, "Congestion control through input rate regula-

- tion," *Proc. IEEE GLOBECOM'89*, pp. 788 - 793, 1989.
- [33] W. Stallings, "*Data and computer communication*," 4nd edition, New York: Macmillan, 1994.
- [34] J. S. Turner and L. F. Wyatt, "A packet network architecture for integrated services," *Proc. GLOBECOM'93*, pp. 405 - 410, 1993.
- [35] W. Wang, T. N. Saadawi and K. Aihara, "Bandwidth allocation for ATM networks," *IEEE ICC*, pp. 439-442, 1990.
- [36] W. Willinger, "Traffic modeling for high-speed networks theory versus practice," *Stochastic Networks*, pp. 5-16, 1994.
- [37] X. Wu, I. Lambadaris, H. Lee and A. R. Kaye, "A comparative study of some leaky bucket network access schemes," *Proc. of ICC'94*, pp. 1586 - 1591, 1994.
- [38] J. Zeng and L. F. Merakos, "Analysis of a priority Leaky Bucket regulator for virtual path traffic in ATM networks," *GLOBECOM'94*, pp.606 - 611, 1994.