1. The first step in the process of creating a new product is to identify a market need. This involves conducting market research to understand what consumers want and what problems they are facing.

 

PIONEERS IN METHYLENE BLUE TESTING SINCE 1974

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

Votre file - Votre référence

Votre file - Votre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Multiplexing, Cell Loss and Traffic Controls in ATM Networks

by

Rui Lin

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

Master of Electrical Engineering

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

May, 1992

© Rui Lin, 1992

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

Author's Address

Author's Address

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 0-315-79812-2

Canada

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

**“Multiplexing, Cell Loss and
Traffic Controls in ATM Networks”**

submitted by

Rui Lin

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

Thesis Co-supervisor

Thesis Co-supervisor

Chairman,

Department of Systems and Computer Engineering

Ottawa-Carleton Institute for Electrical Engineering,
Department of Systems and Computer Engineering
Carleton University

May, 1992

Abstract

The thesis presents a queueing model for an ATM multiplexer with bursty traffic. Analytic solutions of average cell loss probabilities are derived considering traffic descriptors, such as loads in active and idle states, and connection statistics for a simple case and a general case. Solutions are obtained for a set of loss performance descriptors: average cell loss period, non-loss period, average number of cells lost in loss period, loss rate in loss period and loss period variance. Although an assumption of long-burst traffic is made in the cell loss behavior analysis, it is shown that the solution can also provide a good approximation for shorter bursty traffic. An improved approach to traffic control in BISDN networks is suggested.

***To :
My Parents and My Family***

Acknowledgments

I would like to thank my thesis supervisors, Prof. S. A. Mahmoud and A. R. Kaye for their guidance and encouragement during the course of this research. It has been a great opportunity to work under them and to benefit from their deep knowledge of the subject.

I would like to express my gratitude to my parents and my beloved family for their support during my studies and the preparation of this thesis.

I gratefully acknowledge the funding support provided by my thesis supervisors, Carleton University, Telecommunications Research Institute of Ontario, and National Sciences and Engineering Research Council.

Acronyms

ATM	Asynchronous Transfer Mode
STM	Synchronous Transfer Mode
BISDN	Broadband Integrated Service Digital Network
CBR	Continuous Bit Rate
VBR	Variable Bit Rate
i. i. d.	independent identically distributed
CCS	Hundred Call Seconds (per hour)
MX	Multiplexer
SW	Switch
QOS	Quality Of Service
QLT	Queue Length Threshold
HOL_PJ	Head-of-the-Line with Priority Jump
FCFS	First Come First Serve
LB	Leaky Bucket
VLB	Virtual Leaky Bucket
SG_Q	Stop-and-Go Queueing
FIFO	First In First Out
ATMSS	ATM Multiplexer Simulation Software

Table of Contents

Abstract	iv
Acknowledgments	v
Acronyms	vi
List of Figures	x
List of Tables	xii
 Chapter 1 Introduction	 1
1.1 Background and Motivations	1
1.2 Objectives of the Thesis	4
1.3 Outline of the Thesis	4
 Chapter 2 Traffic and Multiplexer Models: Restricted Buffer Size	 6
2.1 Introduction	6
2.2 ATM Traffic Model	7
2.3 Queueing Model of an ATM Multiplexer	9
2.4 Theoretical Results	11
2.4.1 Multiplexing One Group of Bursty Traffic	11
2.4.2 Multiplexing Two Groups of Bursty Traffic	14
2.4.3 Multiplexing Mixed Bursty and CBR Traffic	15
2.5 Simulation and Numerical Results	16
2.5.1 Multiplexing a Group of Bursty Traffic	16
A Simple Case	16
The Case with General ρ_1 and ρ_2	17
Variation of Load ρ_2	18
Variation of ρ_1	22
Variation of ρ_1 and ρ_2	24
Variation of Connection Load	26
Variation of Burstiness Factor	27
2.5.2 Multiplexing Two Different Types of Traffic Groups	31
2.5.3 Multiplexing One Group of Bursty and One Group of CBR Traffic	32
2.6 Conclusions	36

Chapter 3 Extended Multiplexer Model: General Buffer Size Case	37
3.1 Introduction	37
3.2 Revisit of Traffic Model and Analysis	38
3.3 Analysis of the Multiplexer Model	42
3.4 Simulation and Numerical Results	44
Variations of Buffer Size	46
Variations of Burstiness Factor	47
Variations of the Number of Sources	47
3.5 Conclusions	50
 Chapter 4 Cell Loss Process	 51
4.1 Introduction	51
4.2 Types of Traffic Covered by Cell-Loss Analysis	52
4.3 Analysis Of Cell Loss Process	54
4.4 An Object-Oriented Cell Loss Simulator	60
4.4.1 Object-Oriented Programming Preliminary	61
4.4.2 Object-Oriented Design of ATM Multiplexer Simulation Software (ATMSS)	64
Design Objectives	64
Design Guidelines	65
4.4.3 Conceptual Design and Implementation	65
Involving Object-Oriented Concepts	65
The Conceptual Framework	66
4.4.4 Discussion of Object-Oriented ATMSS Design	69
4.5 Simulation Confirmations	70
4.5.1 long-burst traffic	70
4.5.2 High Threshold Voice Traffic	71
4.5.3 Low Threshold Voice Traffic	74
4.6 Numerical Results	74
4.6.1 Variations of Buffer Size	75
4.6.2 Variations of Burstiness Factor	76
4.6.3 Variations of Number of Sources	79
4.6.4 Effect of Traffic Arrival Rate	81
4.6.5 Examination of Consecutive Loss	83
4.7 Discussion And Concluding Remarks	87

Chapter 5 Evaluation of Traffic Control in ATM Networks and a Proposal	92
5.1 Introduction	92
5.2 Various Traffic Congestion Controls and Associated Problems	94
5.2.1 Admission Control	94
5.2.2 Buffer and Queueing Management	97
5.2.3 Traffic Policing	98
5.2.4 Stop-and-Go Queueing	100
5.2.5 Reactive Control	101
5.3 Error Control and Recovery Schemes	102
5.4 A Multi-Paradigm Approach to Control of Integrated Traffic	104
5.4.1 Admission Control	105
5.4.2 Buffer and Queueing Management	105
5.4.3 Traffic Policing	105
5.4.4 Stop-and-Go Queueing	105
5.4.5 Reactive Control	106
5.4.6 Error Control and Recovery Schemes	106
5.5 Concluding Remarks	106
 Chapter 6 Conclusions	 107
6.1 Conclusions	107
6.2 Suggestions for Further Study	108
 References	 109

List of Figures

Figure 2.1	Traffic Models in Various Levels	8
Figure 2.2	Model of an ATM Multiplexer with One Group of Bursty Sources . .	10
Figure 2.3	Models of an ATM Multiplexer with Two Groups of Bursty Sources .	10
Figure 2.4	Model of an ATM Multiplexer with Mixed Bursty and CBR Sources .	12
Figure 2.5	Cell Loss Probability vs Buffer Size: the comparison of analytical results with simulations in C and on QNAP2.	17
Figure 2.6	Cell Loss Probability vs Buffer Size: Analytical Solutions and Simulations. Number of connections $N=20$, $\rho_1=1$ and $\rho_2=0.2$	19
Figure 2.7	Distribution of Number of Cells Entering the Multiplexer: the Effect of Various ρ_2 . Number of connections $N=300$	21
Figure 2.8	Cell Loss Probability against Buffer Size: the Effect of Various ρ_2 . Number of connections $N=300$	21
Figure 2.9	Bandwidth Gain vs Load in Inactive State for a Given Cell Loss Probability. Number of connections $N=300$	22
Figure 2.10	Distribution of Number of Cells Entering the Multiplexer for Different Values of ρ_1 . Number of connections $N=300$	23
Figure 2.11	Cell Loss Probability against Buffer Size: the Effect of ρ_1 . Number of connections $N=300$	23
Figure 2.12	Bandwidth Gain Achieved for a Given ρ_1 and Cell Loss Probability. Number of connections $N=300$	24
Figure 2.13	Distribution of Number of Cells Entering the Multiplexer for Different Values of r_{12} . Number of connections $N=300$	25
Figure 2.14	Cell Loss Probability vs Buffer Size. Number of connections $N=300$. .	26
Figure 2.15	Distribution of Number of Cells Entering the Multiplexer for Different Connection Loads. Number of connections are 300.	27
Figure 2.16	Cell Loss Probability vs Buffer Size for Various Connection Loads. Number of connections $N=300$	28
Figure 2.17	Bandwidth Gain against Connection Load for Different Cell Loss Probability. Number of connections $N=300$	29
Figure 2.18	Distribution of Number of Cells Entering the Multiplexer for Various Burstiness Factors. Number of connections $N=300$	30
Figure 2.19	Cell Loss Probability vs Buffer Size for Different Burstiness Factors. Number of connections $N=20$	30
Figure 2.20	Bandwidth Gain vs Burstiness Factor for Various Cell Loss Probabilities. $T_1=12$ and Number of connections $N=300$	31

Figure 2.21 Cell Loss Probability vs Buffer Size: Analytical Solutions and Simulations. Number of connections $N=20$.	32
Figure 2.22 Cell Loss Probability vs Buffer Size: Analytical Solutions and Simulations. Number of bursty connections $N_b=10$ and number of CB connections $N_c=10$.	33
Figure 3.1 Traffic State Transition Diagrams	40
Figure 3.2 Transition of the Number of Arrival Cells During a Unit Time	43
Figure 3.3 Cell Loss Probability vs Trunk Capacity: $T_1=200$ and $T_2=270$	45
Figure 3.4 Cell Loss Probability vs Trunk Capacity: $T_1=40771$ and $T_2=50000$	45
Figure 3.5 Cell Loss Probability vs Buffer Size: $T_1=200$ and $T_2=270$	46
Figure 3.6 Cell Loss Probability vs Buffer Size: $T_1=40771$ and $T_2=50000$	47
Figure 3.7 Cell Loss Probability vs Trunk Capacity: $T_1=200$	48
Figure 3.8 Cell Loss Probability vs Trunk Capacity: $T_1=50000$	48
Figure 3.9 Cell Loss Probability vs Number of Sources: $T_1=200$ and $T_2=270$	49
Figure 3.10 Cell Loss Probability vs Number of Sources: $T_1=40771$ and $T_2=50000$	49
Figure 4.1 Loss state transition diagram	56
Figure 4.2 Fig. 1 Object Interaction Diagram	69
Figure 4.3 Cell Loss Period Versus Trunk Capacity	71
Figure 4.4 Cell Non-Loss Period Versus Trunk Capacity	72
Figure 4.5 Average Number of Cells Lost In Loss Period	72
Figure 4.6 The Variance Of Cell Loss Period Versus Trunk Capacity	73
Figure 4.7 Cell Loss Rate In Loss Period Versus Trunk Capacity	73
Figure 4.8 Average Loss Rate vs Trunk Capacity	79
Figure 4.9 Average Loss Period vs Trunk Capacity	80
Figure 4.10 Non-loss Period vs Trunk Capacity	81
Figure 4.11 Loss Rate in Loss Period vs Trunk Capacity	82
Figure 4.12 Number of Cells Lost in Loss Period vs Trunk Capacity	83
Figure 4.13 Average Cell Loss Rate vs the Number of Sources	84
Figure 4.14 Average Cell Loss Period vs the Number of Sources	85
Figure 4.15 Average Non-loss Period vs the Number of Sources	86
Figure 4.16 Average Loss Rate in Loss Period vs the Number of Sources	87
Figure 4.17 Average Number of Cells Lost in Loss Period vs the Number of Sources	88
Figure 5.1 ATM Traffic Control Model	93

List of Tables

Table 2.1	Cell Loss Probabilities from Closed Form Solution and Simulations. (* means that the values could only be obtained by longer simulation)	18
Table 2.2	Cell Loss Probabilities from Analytical Solutions and Simulations. $\rho_1=1$ and $\rho_2=0.2$. (* means that the values could only be obtained by longer simulation)	20
Table 2.3	Cell Loss Probabilities from Analytical Solutions and Simulations. (* means that the values could only be obtained by longer simulation)	34
Table 2.4	Cell Loss Probabilities from Analytical Solutions and Simulations. (* means that the values could only be obtained by longer simulation)	35
Table 4.1	Comparison of Simulations and Analytical Results for the High Threshold Voice Case	75
Table 4.2	Comparison of Simulations and Analytical Results for the Low Threshold Voice Case	76
Table 4.3	Cell Loss Performance vs Buffer Size at $M=8$ and $nc=12$. .	77
Table 4.4	Cell Loss Performance vs Buffer Size at $M=8$ and $nc=18$. .	78
Table 4.5	Cell Loss Performance vs Trunk Capacity: Buffer Size= $nc=30$	89
Table 4.6	Cell Loss Performance vs Trunk Capacity: Buffer Size= $nc=40$	89
Table 4.7	Cell Loss Performance vs Number of Sources and Trunk Capacity when Networks Operate at the Level of 10^{-9} Cell Loss Rate	90
Table 4.8	Cell Loss Performance vs Number of Sources and Trunk Capacity when Networks Operate at the Level of 10^{-6} Cell Loss Rate	90

Chapter 1

Introduction

1.1 Background and Motivations

The rapidly increasing demand for high quality and fast multimedia communication services, and the emergence of high-speed telecommunications technologies have provided an impetus for the development of Broadband ISDN (BISDN) networks [1, 2, 3, 4, 5]. BISDN will circumvent two major drawbacks of conventional packet switching networks: separated services and low speed. It is planned as a high-speed, all-purpose, digital network.

BISDN will be capable of supporting a wide range of applications such as voice, low and high speed data, image, and video, with a broad spectrum of traffic types and performance requirements. ATM (Asynchronous Transfer Mode) has finally been accepted as the final transfer mode for BISDN [6]. Traffic in ATM is organized into fixed-size transmission units called cells, each consisting of a header and an information field. These cells are transmitted via a virtual circuit through multiplexers, intermediate switching nodes and demultiplexers; cells belonging to the same virtual circuit are identified by the header. ATM is essentially a connection-oriented technique, though it is envisioned as a basis for supporting all services, connectionless as well as connection-oriented.

The advantages of the ATM multiplexing technique included: 1) flexibility in integrating the transport of connections with a wide range of bandwidth requirements, and in accommodating future applications; 2) potential to attain high bandwidth efficiency by statistically multiplexing bursty traffic flows; 3) minimal delay and variance of delay due to the small fixed size of cell. However, all these advantages can be seriously hindered in the absence of traffic control or with inappropriate traffic control.

Although ATM shares much in common with conventional packet switching at lower rates, for which the subject of traffic control has been studied extensively, and for which several workable control techniques are known, most of these conventional approaches to traffic control do not scale well to the extremely high speeds associated with BISDN. Due to the rich mixture of services and applications anticipated in future ATM networks, traffic control becomes a challenging issue. We cannot immediately resort to separated and complicated traffic controls because the heavy processing of the control functions will force the network to give up its unique high-speed feature. We also cannot simply employ one traffic control and ignore the quality of service (QOS) needs. Efficient traffic controls will therefore be important in this network in order to ensure that sufficient bandwidth and satisfactory performance are provided to all multiplexed connections [7, 8].

In contrast to the traffic control schemes in conventional packet switching networks, ATM traffic control has two key points. One is how to allocate the bandwidth for each call to guarantee the required transfer performance. The other key point is how to control the actual input traffic. In an ATM network, input traffic may change during a call. Even if the bandwidth of a call exceeds the allocated level so that the total load in the network increases unexpectedly, the transfer performance for other calls must be guaranteed. An efficient method of traffic control is to allocate bandwidth for a call at a call set-up stage and to control traffic so that it does not exceed the allocated bandwidth during the call.

There is an increasing amount of research dedicated to ATM traffic controls (ref. to [7]). However, a number of issues are not completely solved. Lack of agreement is delaying the progress of international standardization of BISDN, in particular, the traffic control functions in various layers. Even if international standardization is finished, there are still some issues related to detailed implementation which need to be studied. We observe that the following very important problems are ignored in most of the previous research.

- Cell loss behavior and its implication as the basis for traffic control evaluations. In contrast to the conventional packet switching network, cell loss, instead of transmission

delay, becomes the most critical performance index in a high-speed ATM network. Similar to the probability of average bit error used in the conventional network and communication channel analyses, the probability of average cell loss is employed by most of the previous research as the basis of evaluating interested traffic control schemes. Since the most common traffic in a future high-speed ATM network will be bursty (even those which are not bursty at the beginning in their sources may cluster in the network and become bursty), cell loss will not be an i.i.d. (independent identically distributed) random process any more. Cell loss will be correlated. It will have its own loss and non-loss periods. While the overall average cell loss probability in the entire call active period can be kept very low, cell loss in a particular period may be very large. Average cell loss probability alone cannot provide enough information for the system performance. This unique characteristic of cell loss in an ATM network should be taken into account when studying the traffic control performance. For example, an error control scheme based on the average cell loss probability would not be efficient, therefore not valid in an ATM network. It should be noted that there is not an exact solution for the cell loss process, especially for bursty traffic available so far. Li [9] provides a solution for blocking behavior rather than cell loss pattern. Bondi [10] studies the cell loss patterns of a switch for non-bursty traffic. Schulzrinne and Kurose [11] analyze the time-out loss behavior for a single input stream instead of actual cell loss period.

- An integrated view of traffic controls in ATM networks. Most of the previous research take a one-angle view approach, i.e., focusing on a particular traffic control scheme for a particular type of traffic, without addressing how it combines with other schemes and where and when it is invoked. Here, we are not saying that a one-angle view is not good; rather, we think that it is very important, particularly in understanding, in-depth, various traffic characteristics and associated control features. In my thesis, I also employ this one-angle approach. However, it is time to take a systematic view of the traffic control issue, since a single control scheme is not enough to provide a good quality of service for all kinds of traffic.

The above mentioned major problems motivate the research in this thesis. Other

particular problems associated with a specific traffic and control type will be addressed in the following corresponding chapters in due course.

1.2 Objectives of the Thesis

In this thesis, I concentrate on the performance of an ATM multiplexer queueing model, cell loss process, and an integrated examination of BISDN traffic controls. The objectives of this thesis include:

1. Using a step-by-step approach to establish a more close-to-practice ATM multiplexer queueing model and bursty traffic model intended for data and voice applications.
2. Presenting an analytical approach to analyze cell loss process.
3. Developing simulation software, in particular, object-oriented simulation software for cell loss process, to simulate the performance of an ATM multiplexer queueing model and to verify our analytical results.
4. Using our analytical solution to evaluate ATM multiplexing performance in terms of cell loss, and employing a one-angle-view approach to reexamine some traffic control schemes, in particular, traffic shaping and error control.
5. Integrally evaluating various traffic control features, and identifying a set of traffic control schemes best for BISDN services.

1.3 Outline of the Thesis

The thesis is organized as follows.

Chapter 2 first establishes an ATM bursty traffic model and a simple ATM multiplexer model with buffer size being equal to trunk capacity. A preliminary analytical solution of the average cell loss probability is then obtained for this simple multiplexer model. A simulation software written in C is designed to simulate the simple model and to verify our analytical results. It is noted that the bursty traffic model which considers various loads in active and idle states covers more scenarios than that employed in [12].

Chapter 3 studies the performance of the multiplexer with no limitation on the buffer

size and the trunk capacity. A more general analytical solution of the average cell loss probability is presented. The analytical results are also confirmed by simulations.

Chapter 4 presents an analytical approach to analyze the cell loss process in an ATM multiplexer queueing model. Analytical solutions for the lengths of cell loss and non-loss periods, loss period variance, average number of cells lost in loss period, and loss rate in loss period are provided. An object-oriented cell loss process simulator is developed and used to run simulations to confirm our analytical results.

Chapter 5 contains an integrated examination of various traffic control functions. In particular, admission control, traffic shaping, and error control are discussed from the point of view of improving cell loss performance. A set of best-fit traffic control features are identified to be applied to future ATM networks.

Chapter 6 contains some discussion and concluding remarks.

Chapter 2

Traffic and Multiplexer Models: Restricted Buffer Size

2.1 Introduction

A BISDN network consists of various multiplexing and switching elements, in which the ATM multiplexer plays a fundamental role in network performance evaluations. A switching element may be analyzed using the model of a multiplexer, since each output link in a switch can be represented as a multiplexer which concentrates all the inputs of the switching element. In an ATM multiplexer, traffic from all the input sources is usually concentrated on a higher-speed output trunk. Due to the complexities of the traffic carried in the network, the analysis, especially the theoretical analysis of the multiplexer queueing performance, is not an easy task.

Much previous analytical research has been given over to an ATM multiplexer queueing model with either a simplified configuration of the multiplexer or an idealized type of traffic. An ATM multiplexer model, with specific buffer size (as will be mentioned below) and homogeneous and non-identical traffic sources, has been studied in [12]. In [12], a simple two-state traffic model is used without considering various loads in the active and idle state. A general buffer size case is analyzed in [13]. However, a group of homogeneous traffic sources is still assumed. Moreover, the statistics of the connections are ignored and the computational time complexity of the analytical model increases drastically, especially when a great number of connections are considered. However, it is still more cost-effective than simulations. A queueing analysis for multiplexing mixed continuous-bit-rate (CBR) and bursty traffic is performed in [14]. Instead of using the realistic CBR traffic, a random traffic model [14] has been employed to approximate CBR traffic. This leads to a pessimistic analysis.

To overcome the above-mentioned drawbacks, and for the simplicity of the preliminary research, this chapter first studies a simplified ATM multiplexer in which buffer size is equal to trunk capacity which is, in turn, defined as the maximum number of cells the multiplexer output trunk can transmit during the cell holding time (i.e., cell duration) at source. In this model, no cell queueing appears. The cell delay fluctuation introduced by the queueing delay is therefore eliminated, and hence the total cell delay is reduced as well. As will be seen, the major advantage of employing this simple model is that it makes the analysis of more complicated types of traffic possible. In the analysis of this chapter, homogeneous and heterogeneous traffic sources are considered.

2.2 ATM Traffic Model

ATM networks provide an integrated service for all kinds of traffic ranging from variable bit rate (VBR) to CBR. In VBR traffic, typically, the bursty traffic is the main cause of the difficulties in guaranteeing the required quality of service by the network, and is the cause of network congestion. Our studies will focus on the bursty traffic, in particular, the bursty data including image data traffic and the bursty voice traffic. The following is a description of our bursty traffic model.

At the call level, the source is either connected or disconnected. In Fig. 2.1(a), the connection period and disconnected period are assumed to be independent, identically-distributed random variables which follow a geometric distribution. This assumption is valid for interactive data transfer and voice calls. It is also suitable in the case of video transmission [15]. The probability of connection can be found to be

$$P_c = \frac{T_c}{T_c + T_d} \quad (2.1)$$

where T_c and T_d are the average values of the connection period and disconnected period, respectively.

At the ATM cell level, input traffic is bursty and can be assumed to have two states, S_1 and S_2 , as shown in Fig. 2.1(b) [13]. The states S_1 and S_2 are called active and

2.1(a)

2.1(b)

2.1(c) Peak value $\rho = 1$ 2.1(d) Peak value $\rho < 1$

Figure 2.1 Traffic Models in Various Levels

inactive state, respectively. Input line cell holding time, T_{in} , is defined as a unit time. In the state S_1 , cell arrives at rate ρ_1 , whereas the cell arrival rate is ρ_2 ($\rho_1 > \rho_2$) in the state S_2 . The periods of S_1 and S_2 are assumed to be geometrically distributed, with mean values of T_1 and T_2 unit times, respectively. When $\rho_1=1$ and $\rho_2=0$, the model is reduced to the one used in [12].

This model can describe the interactive data traffic and voice call traffic. In the interactive data traffic, the state S_1 corresponds to the active data transfer state, whereas

the state S_2 corresponds to the idle period in the data transfer. In a voice telephone call, the state S_1 refers to talk spurt and the state S_2 refers to pause in human voice. For compressed video communication, the frame length corresponding to the state S_1 and the inter-frame periods corresponding to the state S_2 , may also be assumed to be approximately geometrically distributed [15]. ρ_2 is, in general, not zero due to some non-bursty information transfer between frames. For more accurate modeling of video traffic, a discrete-state continuous-time Markov process, with bath arrivals, may be applied [16].

If y sources are connected, the probability that the number of active input lines is k in a unit time is

$$P_a(k) = \binom{y}{k} P_1^k P_2^{y-k} \quad (2.2)$$

where P_i ($i=1, 2$) denotes the probability that an input line state is S_i with the expressions

$$P_1 = \frac{T_1}{T_1 + T_2} \quad \text{and} \quad P_2 = \frac{T_2}{T_1 + T_2} \quad (2.3)$$

2.3 Queueing Model of an ATM Multiplexer

An ATM multiplexer will multiplex all kinds of traffic with various complicated characteristics. It is impossible as well as impractical to exactly model every aspect of the traffic sources. For the sake of bringing out the interesting and important features, and for the simplicity of analysis, three scenarios of an ATM multiplexer model shown in Fig.2.2, Fig. 2.3 and Fig. 2.4 are considered.

In Fig. 2.2, a group of homogeneous traffic sources— N inputs with identical traffic characteristics (peak rate R , T_1 , T_2 , ρ_1 and ρ_2)—are connected to the input of the multiplexer. The multiplexing node uses shared cell buffer on the output. It collects from each input periodically and then transmits the cells on the output trunk. If R is the peak rate of the traffic sources in bits/second and L is the length of the cell in bits, each input (when transmitting) generates one cell every $T_{in}=L/R$ seconds. It is assumed that

Figure 2.2 Model of an ATM Multiplexer with One Group of Bursty Sources

Figure 2.3 Models of an ATM Multiplexer with Two Groups of Bursty Sources

this multiplexer model operates synchronously with the input line cell holding time T_{in} . Each cell occupies $T_{out}=L/C$ seconds on the output trunk where C is the trunk speed in bits per second. We will first consider the case that buffer size $Q=T_{in}/T_{out}$. If the number of simultaneously transmitting input sources is less than or equal to Q , there will not be cell loss, otherwise there will be cell loss.

In Fig. 2.3, two groups of bursty traffic sources are considered for the simplicity of the analysis. It is assumed that the multiplexer sample time T_{in} is the cell generation period of the highest peak bit rate source. For example, consider that two types of source, s_1 and s_2 are multiplexed. If the source s_2 bit rate is twice that of s_1 , the multiplexer samples cells every cell period T_{in} and gets one cell in each period from the s_2 source while it gets one cell every alternate period from the s_1 source, provided both are transmitting. For simplicity of calculation, it is assumed that these two traffic groups have the same traffic rates so that each source may transmit one cell in a unit time. Each group has different active period T_1 , inactive period T_2 , and arrival rates ρ_1 and ρ_2 .

Fig. 2.4 is a model of ATM switching of mixed bursty and CBR traffic. In contrast to the approximate treatment in [14], the realistic CBR traffic is employed here. For the sake of calculation simplicity, the cell holding times on the bursty traffic sources and on the CBR traffic sources are assumed to be the same.

2.4 Theoretical Results

In this section, the analytical solutions of cell loss probabilities for three scenarios are derived. As we will see, the assumption that buffer size is equal to trunk capacity simplifies the analyses, the reason being that the buffer is always completely cleared in each time unit.

2.4.1 Multiplexing One Group of Bursty Traffic

We first consider the scenario that a group of bursty traffic with identical traffic characteristics is multiplexed by the multiplexer, as shown in Fig. 2.2. The probability

Figure 2.4 Model of an ATM Multiplexer with Mixed Bursty and CBR Sources

of y calls connected out of N links is shown to be

$$\begin{aligned}
 Prob(y/N) &= \binom{N}{y} P_c^y (1 - P_c)^{N-y} \\
 &= \binom{N}{y} \frac{\left(\frac{T_c}{T_d}\right)^y}{\left(1 + \frac{T_c}{T_d}\right)^N} .
 \end{aligned} \tag{2.4}$$

The probability of x sources being active and y calls being connected is

$$Prob(x, y) = \frac{\binom{y}{x} \left(\frac{T_c}{T_d}\right)^x \binom{N}{y} \left(\frac{T_c}{T_d}\right)^y}{\left(1 + \frac{T_c}{T_d}\right)^y \left(1 + \frac{T_c}{T_d}\right)^N} , \quad x \leq y . \tag{2.5}$$

If the number of active sources is x and the number of connections is y , then m cells will arrive in a unit time with the conditional probability

$$Pr(m/(x, y)) = \sum_{j=0}^N \binom{x}{j} \rho_1^j (1-\rho)^{x-j} \binom{y-x}{i} \rho_2^i (1-\rho)^{y-x-i}, \quad (2.6)$$

$$i = m - j \quad (0 \leq i \leq y - x).$$

Hence, the probability distribution of the number of arriving cells in a unit time is given by

$$Pr_1(m) = \sum_{x=0}^N \sum_{y=x}^N Prob(x, y) Pr(m/(x, y)). \quad (2.7)$$

The probability of k cell loss, which is the probability of more than Q cell arrival in a unit time, is obviously

$$Prob(k \text{ cell loss}) = Pr_1(Q + k). \quad 1 \leq k \leq N. \quad (2.8)$$

The probability of average cell loss in a unit time, which can be defined as the average number of cells lost over the average number of cells arrived in a unit time, is found to be

$$P_{loss1} = \frac{\sum_{m=Q+1}^N Pr_1(m)(m-Q)}{\sum_{m=0}^N Pr_1(m)m}. \quad (2.9)$$

It should be noted that expression (2.9) is, in fact, more general than the solution obtained in [12]. When the traffic loads take a particular value, i.e., $\rho_1=1$, $\rho_2=0$, expression (2.9) is reduced to the one in [12].

In general, ρ_1 is not equal to one and ρ_2 is not equal to zero. For instance, in video transmission, the period between one scene changing to the other corresponds to an idle state. This idle period is not absolutely idle since a small portion of one scene may

vary and may need to be transmitted. However, the load in an idle state is much lower than that in an active state.

2.4.2 Multiplexing Two Groups of Bursty Traffic

We assume that the two groups of traffic have the same priority. For the case of different priorities, the solution will be easier to obtain from the following derivation.

The probability of z cells entering the multiplexer can be expressed as

$$Pr_2(z) = Pr_2(z = x_1 + x_2) , \quad (2.10)$$

where x_1 cells are from the first group of bursty sources and x_2 from the second. Using the results obtained in (2.5) and (2.7) the above expression can also be derived as

$$\begin{aligned} Pr_2(z) &= \sum_{x_2=0}^{\min(z, N_{b2})} Pr_1(x_1 = z - x_2 / x_2) Pr(x_2) \\ &= \sum_{x_2=0}^{\min(z, N_{b2})} \sum_{i=0}^{N_{b1}} \sum_{x_1=i}^{N_{b1}} \frac{\binom{x_1}{i} \left(\frac{T_{11}}{T_{21}}\right)^i \binom{N_{b1}}{x_1} \left(\frac{T_{c1}}{T_{d1}}\right)^{x_1}}{\left(1 + \frac{T_{11}}{T_{21}}\right)^{x_1} \left(1 + \frac{T_{c1}}{T_{d1}}\right)^{N_{b1}}} Pr[z - x_2 / (i, x_1)] \times \\ &\quad \times \sum_{j=0}^{N_{b2}} \sum_{x_2=j}^{N_{b2}} \frac{\binom{x_2}{j} \left(\frac{T_{12}}{T_{22}}\right)^j \binom{N_{b2}}{x_2} \left(\frac{T_{c2}}{T_{d2}}\right)^{x_2}}{\left(1 + \frac{T_{12}}{T_{22}}\right)^j \left(1 + \frac{T_{c2}}{T_{d2}}\right)^{N_{b2}}} Pr[x_2 / (j, x_1)] , \end{aligned} \quad (2.11)$$

in which T_{11} and T_{21} are the average active length and idle length of the first group of traffic, and T_{12} and T_{22} are of the second group of traffic. Similarly, T_{c1} and T_{d1} are the average connected length and disconnected length of the first group of traffic, and T_{c2} and T_{d2} are of the second group of traffic. The probability of k cell loss is

$$Prob(k \text{ cell loss}) = Pr_2(Q + k) \quad 1 \leq k \leq N_{b1} + N_{b2} . \quad (2.12)$$

The probability of average cell loss in an unit time is

$$P_{loss2} = \frac{\sum_{z=Q+1}^{N_{b1}+N_{b2}} Pr_2(z)(z-Q)}{\sum_{z=0}^{N_{b1}+N_{b2}} Pr_2(z)z} . \quad (2.13)$$

2.4.3 Multiplexing Mixed Bursty and CBR Traffic

Again, we consider only the same priority case.

Following the above steps and using the expressions in (2.5) and (2.7), we show that the probability of z cells entering the multiplexer is

$$\begin{aligned} Pr_3(z) &= \sum_{x_2=0}^{\min(z, N_c)} Pr_1(x_1 = z - x_2 / x_2) Pr_1(x_2) \\ &= \sum_{x_2=0}^{\min(z, N_c)} \sum_{i=0}^{N_{b1}} \sum_{x_1=i}^{N_{b1}} \frac{\binom{x_1}{i} \left(\frac{T_{d1}}{T_{s1}}\right)^i \binom{N_{b1}}{x_1} \left(\frac{T_{s1}}{T_{d1}}\right)^{x_1}}{\left(1 + \frac{T_{d1}}{T_{s1}}\right)^{x_1} \left(1 + \frac{T_{s1}}{T_{d1}}\right)^{N_{b1}}} \times \\ &\quad Pr[z - x_1 / (i, x_1)] \frac{\binom{N_c}{x} \left(\frac{T_{s2}}{T_{d2}}\right)^{x_2}}{\left(1 + \frac{T_{s2}}{T_{d2}}\right)^{N_c}} . \end{aligned} \quad (2.14)$$

in which the definition of various variables is similar to those declared before this subsection. The probability of k cell loss in an unit time is

$$Prob(k \text{ cell loss}) = Pr_3(Q + k) \quad 1 \leq k \leq N_{b1} + N_c . \quad (2.15)$$

The probability of average cell loss in an unit time is

$$P_{loss3} = \frac{\sum_{z=Q+1}^{N_{b1}+N_c} Pr_3(z)(z-Q)}{\sum_{z=0}^{N_{b1}+N_{b2}} Pr_3(z)z} . \quad (2.16)$$

2.5 Simulation and Numerical Results

2.5.1 Multiplexing a Group of Bursty Traffic

A Simple Case

To validate our simulation as well as the theoretical results, we first consider a simple case where the load in bursty state $\rho_1=1$ and the load in inactive state $\rho_2=0$. This is a special case for our model and is exactly the same as the model studied in [12]. To compare our numerical results from the analytical approach and from simulations, several programs written in C are developed.

An example of analytical results confirmed by simulation in C and on QNAP2 is shown in Figure 2.5 and in Table 2.1, where the number of connections to the multiplexer is 20. In this example, $T_c=18000$, $T_d=36000$ (which is equivalent to the traffic intensity of 12 CCS in [12]), $T_1=12$ and $T_2=18$ (i.e., a bursty factor of 2.5) are assumed for comparison. It should be noted that the analytical results and the simulation results on QNAP2 are identical to those presented in [12].

Although we are not doing output data and confidence interval analysis in our C simulations (the emphasis of this thesis is not on simulations), the comparison with QNAP2 (with 95% confidence level) and the simulation results in [12] shows the correctness of our C simulations.

From Figure 2.5 and Table 2.1, we can observe that the cell loss probability reaches one when buffer size is zero. With the increase in buffer size (trunk capacity), the cell loss probability drops dramatically. Therefore, buffer size, i.e., trunk capacity, has a significant effect on the cell loss probability.

Figure 2.5 Cell Loss Probability vs Buffer Size: the comparison of analytical results with simulations in C and on QNAP2.

The Case with General ρ_1 and ρ_2

In reality, ATM traffic in the network does not always conform to its claim at admission. Cells tend to cluster each other in the network. This will, to some extent, alter the loads in both the bursty and inactive states. In our analytical model, this scenario is taken into account.

To verify the theoretical approach, simulation in C is run and the results are presented along with the analytical ones, in Figure 2.6 and Table 2.2. In Figure 2.6 and Table 2.2, $T_c=18000$, $T_d=36000$, $T_1=12$ and $T_2=18$ are assumed. The load in bursty state ρ_1 is 1 and in inactive state ρ_2 is 0.2. In comparing Figure 2.5 and Figure 2.6, we note that the extra load introduced in the inactive state increases the cell loss probability. This also coincides with what we expect by deduction. More significantly, Figure 2.6 and Table 2.2 further show the correctness of the analytical solutions.

Buffer Size (Trunk Capacity) (N=20)	Probability of Cell Loss		
	Analytical Results	Simulation in C	Simulation on QNAP2 (confidence level=0.95)
0	1.000e-0	1.000e-0	1.000e-0
1	0.646e-0	0.649e-0	0.643e-0
2	0.358e-0	0.359e-0	0.355e-0
3	0.167e-0	0.170e-0	0.165e-0
4	0.653e-1	0.652e-1	0.643e-1
5	0.212e-1	0.219e-1	0.212e-1
6	0.575e-2	0.573e-2	0.588e-2
7	0.130e-2	0.131e-2	0.139e-2
8	0.247e-3	0.253e-3	0.292e-3
9	0.392e-4	0.424e-4	0.446e-4
10	0.521e-5	0.613e-5	*
11	0.576e-6	0.748e-6	*
12	0.525e-7	*	*
13	0.391e-8	*	*
14	0.234e-9	*	*
15	0.110e-10	*	*

Table 2.1 Cell Loss Probabilities from Closed Form Solution and Simulations.

(* means that the values could only be obtained by longer simulation)

Variation of Load ρ_2

The probability of having x number of cells entering simultaneously into the multiplexer, $P(x)$, is plotted in Figure 2.7, with the traffic load in the bursty state being fixed $\rho_1=1$, and the traffic load in the inactive state ρ_2 varying from 0.0 to 0.9. The multiplexer is assumed to have 300 connections. All other parameters are kept constant,

Figure 2.6 Cell Loss Probability vs Buffer Size: Analytical Solutions and Simulations. Number of connections $N=20$, $\rho_1=1$ and $\rho_2=0.2$.

i.e., $T_c=18000$, $T_d=36000$, $T_1=12$ and $T_2=18$. The number of cells corresponding to the peaks of $P(x)$ in Figure 2.7 are nothing but the average number of cells transmitting simultaneously. When the traffic load in the inactive state is increased, the average number of cells entering at the same time increases as well.

From Figure 2.7, one can see that the distribution of the number of cells entering is close to Normal distribution. This can be justified by the Central Limit Theorem because the random variable, the number of cells entering, is a result of the sum of many identical independent variables, the number of cells transmitted from each connection.

Cell loss probability, $\text{Prob}(\text{loss})$, is plotted in Figure 2.8 for various traffic loads in the inactive state. The $\text{Prob}(\text{loss})$ reduces as the buffer size increases. For lower values of ρ_2 , the buffer size required to achieve a given cell loss probability is lower.

Buffer Size (Trunk Capacity) (N=20)	Probability of Cell Loss	
	Analytical Results	Simulations in C
0	1.000e+00	1.000e+00
1	7.179e-01	7.184e-01
2	4.628e-01	4.633e-01
3	2.611e-01	2.613e-01
4	1.268e-01	1.267e-01
5	5.247e-02	5.228e-02
6	1.842e-02	1.830e-02
7	5.468e-03	5.437e-03
8	1.371e-03	1.379e-03
9	2.895e-04	3.008e-04
10	5.133e-05	5.70e-05
11	7.598e-06	9.0e-06
12	9.314e-07	1.2e-06
13	9.346e-08	*
14	7.547e-09	*
15	4.783e-10	*
16	2.292e-11	*

Table 2.2 Cell Loss Probabilities from Analytical Solutions and Simulations. $\rho_1=1$ and $\rho_2=0.2$. (* means that the values could only be obtained by longer simulation)

Figure 2.9 shows Bandwidth Gain (defined as the ratio of the sum of input peak rates of all the connections to the multiplexer trunk capacity) against Load in the inactive

Figure 2.7 Distribution of Number of Cells Entering the Multiplexer: the Effect of Various ρ_2 . Number of connections $N=300$.

Figure 2.8 Cell Loss Probability against Buffer Size: the Effect of Various ρ_2 . Number of connections $N=300$.

state for different values of cell loss probabilities. It can be noted that the bandwidth

Figure 2.9 Bandwidth Gain vs Load in Inactive State for a Given Cell Loss Probability. Number of connections $N=300$.

gain decreases as the load increases. The bandwidth gain is higher if the acceptable cell loss probability is higher.

Variation of ρ_1

Figure 2.10 shows the probability of having x number of cells entering simultaneously into the multiplexer for various traffic loads in bursty state ρ_1 . The traffic load in the inactive state ρ_2 is fixed. The multiplexer is assumed to have 300 connections. All other parameters are again kept constant, i.e., $T_c=18000$, $T_d=36000$, $T_1=12$ and $T_2=18$. It is clear that when the traffic load in the bursty state is increased, the average number of cells entering at the same time increases.

Figure 2.10 Distribution of Number of Cells Entering the Multiplexer for Different Values of ρ_1 . Number of connections $N=300$.

Figure 2.11 Cell Loss Probability against Buffer Size: the Effect of ρ_1 . Number of connections $N=300$.

Figure 2.12 Bandwidth Gain Achieved for a Given ρ_1 and Cell Loss Probability. Number of connections $N=300$.

Figure 2.11 shows the Prob(loss) curves vs buffer size for various ρ_1 of 1, 0.8, 0.6, 0.4, 0.08. The traffic load in the inactive state ρ_2 is fixed at zero. For a greater bursty load, the trunk capacity required to achieve the same cell loss probability is higher.

In Figure 2.12, bandwidth gain is plotted against traffic load in the bursty state. The higher the traffic load in the bursty state, the lower the bandwidth gain. It is also obvious that the bandwidth gain is higher if higher cell loss probability can be permitted.

Variation of ρ_1 and ρ_2

To analyze the effect of varying ρ_1 and ρ_2 on the multiplexer performance, one first

Figure 2.13 Distribution of Number of Cells Entering the Multiplexer for Different Values of r_{12} . Number of connections $N=300$.

defines two parameters. One is called the offered load at the packet level given by

$$\rho_0 = \frac{T_1 \rho_1 + T_2 \rho_2}{T_1 + T_2}, \quad (2.17)$$

and the other is called the burst-idle load ratio given by

$$r_{12} = \frac{\rho_1}{\rho_2}. \quad (2.18)$$

In the analysis, we will keep ρ_0 constant, for example $\rho_0=0.44$, and vary ρ_1 and ρ_2 , or r_{12} . Figure 2.13 gives $P(x)$ for various burst-idle load ratios (1.8, 2.625, 4, 6.75, 14.0). It is clear that the varying of r_{12} does not affect $P(x)$. We can expect that the change of ρ_0 with r_{12} being fixed is going to influence $P(x)$.

Figure 2.14 shows cell loss probability against buffer size for various r_{12} . One observes that r_{12} does not affect $\text{Prob}(\text{loss})$ as long as ρ_0 is kept constant. If ρ_0 is varied

Figure 2.14 Cell Loss Probability vs Buffer Size. Number of connections N=300.

the cell loss probability will also be changed. The higher the ρ_0 , the higher the cell loss probability.

For bandwidth gain, we can also infer that this will not be affected by the change of r_{12} .

Variation of Connection Load

To describe the traffic intensity at the call level, we define a so-called connection load as

$$\rho_c = \frac{T_c}{T_c + T_d}. \quad (2.19)$$

In order to evaluate the effect of the connection load on the multiplexer performance we vary ρ_c and fix the other parameters, i.e., $T_1=12$, $T_2=18$, $\rho_1=0.8$ and $\rho_2=0.2$.

The probability of having x number of cells entering is plotted in Figure 2.15 for connection loads of 0.25, 0.56, 0.75, 0.85 and 0.90. The multiplexer has 300 connections.

Figure 2.15 Distribution of Number of Cells Entering the Multiplexer for Different Connection Loads. Number of connections are 300.

From Figure 2.15, one can observe that with the increase of connection load the average number of cells entering also increases.

Figure 2.16 shows cell loss probability vs buffer size for various connection loads. Prob(loss) reduces as the buffer size is increased. For lower values of connection load, the buffer size required to maintain a given cell loss probability is smaller.

Bandwidth gain against connection load to achieve different cell loss probabilities is plotted in Figure 2.17. It is clear that bandwidth gain is higher if higher cell loss probability is acceptable.

Variation of Burstiness Factor

The burstiness factor is defined as the ratio of the peak source rate to the average

Figure 2.16 Cell Loss Probability vs Buffer Size for Various Connection Loads. Number of connections $N=300$.

source rate, which can also be expressed as

$$b_f = \frac{T_1 + T_2}{T_1}. \quad (2.20)$$

Continuous bit-oriented traffic has a burstiness factor of one, whereas very bursty sources have higher burstiness factors if T_2 is fixed. However, if T_1 is fixed, a higher burstiness factor does not mean higher bursty source traffic. Asynchronous multiplexers make use of the burstiness of sources being multiplexed to achieve higher bandwidth gains which cannot be obtained by synchronous time-division multiplexing.

The probability of x number of cells transmitting simultaneously for various burstiness factors of 1.05, 1.5, 20, 205, 1500 with $T_1=12$, is shown in Figure 2.18. The other

Figure 2.17 Bandwidth Gain against Connection Load for Different Cell Loss Probability. Number of connections $N=300$.

parameters remain constant, i.e., $T_c=120$, $T_d=240$, $\rho_1=0.8$ and $\rho_2=0.2$. It is observed that with the decrease of the burstiness factor, the average number of cells entering increases.

Figure 2.19 shows cell loss probability vs buffer size for different burstiness factors. It is noted that the buffer size required to guarantee the same cell loss probability is small due to the smaller burst lengths compared with the inactive lengths.

Bandwidth gain vs burstiness factor is plotted in Figure 2.20 for different cell loss probabilities. It is clear that only the increase of the burstiness factor in the low value portion of the horizontal axis explicitly increases the bandwidth gain. The bandwidth gain will reach its limit quickly as the burstiness factor is increased to a certain value. It is not always the case that the higher the burstiness factor, the higher the bandwidth gain. In addition, it should be noted that if a higher cell loss probability is allowed, the bandwidth gain will also be higher.

Figure 2.18 Distribution of Number of Cells Entering the Multiplexer for Various Burstiness Factors. Number of connections $N=300$.

Figure 2.19 Cell Loss Probability vs Buffer Size for Different Burstiness Factors. Number of connections $N=20$.

Figure 2.20 Bandwidth Gain vs Burstiness Factor for Various Cell Loss Probabilities. $T_1=12$ and Number of connections $N=300$.

2.5.2 Multiplexing Two Different Types of Traffic Groups

In this subsection, we present only the results justifying the analytical solutions.

In particular, we consider two different types of traffic groups, with each group having an equal number of connections of 10. Other parameters, except buffer size, are kept unchanged. The parameter values of the first type of traffic group are selected as: connection length $T_{c1}=120$, disconnected length $T_{d1}=240$, bursty length $T_{11}=12$, inactive length $T_{21}=18$, bursty load $\rho_{11}=1$ and inactive $\rho_{21}=0$. The second type of traffic group has connection length $T_{c2}=120$, disconnected length $T_{d2}=240$, bursty length $T_{12}=18$, inactive length $T_{22}=18$, bursty load $\rho_{12}=0.5$ and inactive load $\rho_{22}=0.2$.

Figure 2.21 and Table 2.3 show cell loss probability against various buffer sizes for analytical and simulation results. It is clear that the analytical results are correct. It is also noted that the cell loss probability will be lower if the buffer size is larger.

Figure 2.21 Cell Loss Probability vs Buffer Size: Analytical Solutions and Simulations. Number of connections $N=20$.

2.5.3 Multiplexing One Group of Bursty and One Group of CBR Traffic

In this subsection, we also give the results showing the validation of analytical approach by simulations. The total number of connections considered are 20, with 10 for each group of connections.

Figure 2.22 and Table 2.4 present an example of analytical results confirmed by simulations. In this example, the buffer size is varied and the other parameters are kept constant, i.e., the connection length of bursty traffic $T_{c1}=120$, the disconnected length of bursty traffic $T_{d1}=240$, bursty packet length $T_1=12$, inactive packet length $T_2=18$, load in the bursty state $\rho_1=0.5$, load in the inactive state $\rho_2=0.2$; the connection length of CB traffic $T_{c2}=120$, and the disconnected length of CB traffic $T_{d2}=240$. From Table 2.4, one can see that the analytical results are confirmed very well by simulation at

Figure 2.22 Cell Loss Probability vs Buffer Size:
Analytical Solutions and Simulations. Number of bursty
connections $N_b=10$ and number of CB connections $N_c=10$.

higher cell loss rates. To obtain greater accuracy for much lower loss probabilities, a considerably longer run of simulation would be necessary. We do not attempt to pursue this in the analysis.

It is also noted that the larger the buffer size, the lower of the cell loss probability.

Buffer Size (Trunk Capacity) (N=20)	Probabilty of Cell Loss	
	Analytical Results	Simulations in C
0	1.000e+00	1.000e+00
1	6.277e-01	6.274e-01
2	3.344e-01	3.340e-01
3	1.485e-01	1.481e-01
4	5.466e-02	5.432e-02
5	1.668e-02	1.649e-02
6	4.234e-03	4.156e-03
7	8.965e-04	8.718e-04
8	1.587e-04	1.522e-04
9	2.347e-05	2.18e-05
10	2.899e-06	2.6e-06
11	2.975e-07	*
12	2.520e-08	*
13	1.742e-09	*
14	9.670e-11	*
15	4.20e-12	*
16	1.3e-13	*

Table 2.3 Cell Loss Probabilities from Analytical Solutions and Simulations.

(* means that the values could only be obtained by longer simulation)

Buffer Size (Trunk Capacity) ($N_b=10$ and $N_c=10$)	Probability of Cell Loss	
	Analytical Results	Simulations in C
0	1.000e+00	1.000e+00
1	7.740e-01	0.774e-01
2	5.559e-01	0.556e-01
3	3.606e-01	0.361e-01
4	2.059e-01	0.206e-01
5	1.014e-01	0.102e-01
6	4.241e-02	4.336e-02
7	1.490e-02	1.558e-02
8	4.364e-03	4.729e-03
9	1.058e-03	1.228e-03
10	2.110e-04	2.805e-04
11	3.435e-05	5.72e-05
12	4.524e-06	8.5e-06
13	4.763e-07	*
14	3.942e-08	*
15	2.502e-09	*
16	1.174e-10	*
17	3.83e-12	*

Table 2.4 Cell Loss Probabilities from Analytical Solutions and Simulations.

(* means that the values could only be obtained by longer simulation)

2.6 Conclusions

From the numerical studies carried out in this chapter we can conclude that:

1. The analytical approach is verified because the results are confirmed very well by simulation.
2. Trunk capacity, as well as some traffic parameters, have significant influence on cell loss probability. In particular, the traffic loads in the bursty and inactive states play an important role in affecting cell loss probability. This may suggest their possible use for traffic control. For example, if a traffic shaper is used at the edge of ATM network to reduce the load during the active state, a low cell loss probability is then obtained.

Chapter 3

Extended Multiplexer Model: General Buffer Size Case

3.1 Introduction

This chapter continues the studies in Chapter 2. As we have mentioned in the previous chapter, the simple ATM multiplexer imposes a minimum queuing delay for cells at the network nodes. However, in order to obtain a smaller cell loss rate, either fewer connections or higher trunk capacity are required. Bandwidth gain is therefore reduced. To avoid this loss, one alternative is to increase buffer size under the constraint of a cell delay requirement. This will require a more general ATM queuing model without a special value of buffer.

In this chapter a more general ATM multiplexer queuing model is examined. We present a theoretical approach to analyze average cell loss probabilities when the buffer size of an ATM multiplexer is finite and has a general value. We only consider one group of bursty traffic here. The correlation characteristics [11] of the bursty traffic model are taken into account. The consideration of correlated traffic and a general size of buffer makes dealing with a two-dimensional state transition problem inevitable, and hence increases the complexity of the analysis. At the outset of our analysis we shall assume that an equilibrium solution exists.

It should be noted that the focus of our analysis is on the cell loss performance rather than on the cell delay. Since the ATM multiplexer speed is very high and the buffer we are going to investigate is not very large, we can reasonably assume that the cell loss performance overwhelms the cell delay.

3.2 Revisit of Traffic Model and Analysis

The traffic model employed here is shown in Figure 2.1. If denotes by S_1^i the connection state and S_2^i the disconnection state, we find the probabilities of transition from state S_1^i to state S_j^i , i.e., g_{ij} as follows

$$g_{11} = 1 - \frac{1}{T_c}, \quad g_{12} = \frac{1}{T_c}, \quad g_{21} = \frac{1}{T_d}, \quad \text{and} \quad g_{22} = 1 - \frac{1}{T_d}. \quad (3.1)$$

The state transition diagram for each connection at call level is presented in Figure 3.1(a).

To consider the state transition probabilities for multiple connections at call level we first make an approximation similar to that made by several researchers ([9] and [17]). That is, the process of connection and disconnection can be approximated by a birth-death process. The reason for this is that both connection and disconnection periods are substantially long and it is reasonable to assume that at most only one of the connections may change states in a unit time, which will be defined in due course. For instance, in the voice case, the average length of the call connection is 180 seconds [12]. If the source speed is 64 kbs, the length of the connection is approximately 27169 cells. The resulting probability that a call which is now active becomes idle at the next unit time is only 3.68×10^{-5} . The probability that more than one call change their state simultaneously at the next time slot is much smaller. Therefore, that probability can be neglected. Upon this approximation, the chain can be formed into a quasi-birth-death type which significantly simplifies the numerical analysis. See Figure 3.1(b), the state transition diagram, in which the state is defined as the number of sources connected. The only possible transitions are those to neighboring states.

The state transition probabilities of the number of sources connected are easily obtained in the following way. Assume that the current state is in the state E_i , meaning i sources connected. If one of the i connected sources becomes disconnected and all the other sources remain in the same state, the system changes its state to E_{i-1} . This happens with the probability $C_i^{-1} g_{12} (1 - g_{12})^{i-1} (1 - g_{21})^{nc-i}$. If one of the $nc-i$ disconnected sources becomes connected, and all other sources remain in the same state, the system goes to a

new state E_{i+1} with the probability of $C_{nc-1}^{-1} g_{21}(1-g_{21})^{nc-i-1}(1-g_{12})^i$. Following the above discussion with a consideration of the boundary cases where the system is in E_0 and E_{nc} , we obtain the following expression for the state transition probability $h_{i,j}$ from E_i to E_j :

$$h_{i,j} = \begin{cases} ig_{12}(1-g_{12})^{i-1}(1-g_{21})^{nc-i} & 0 \leq j = i-1 \leq nc-1 \\ (nc-i)g_{21}(1-g_{21})^{nc-i-1}(1-g_{12})^i & 1 \leq j = i+1 \leq nc \\ 1 - h_{i,i-1} - h_{i,i+1} & 1 \leq i = j \leq nc-1 \\ 1 - h_{0,1} & i = j = 0 \\ 1 - h_{nc,nc-1} & i = j = nc \\ 0 & \text{otherwise} \end{cases} \quad (3.2)$$

At the ATM cell level, input traffic is bursty and can be assumed to have two states, S_1 and S_2 , as shown in Fig. 2.1(b) [13]. The states S_1 and S_2 are called active and inactive states, respectively. Input line cell holding time T_{in} , whose relationship with trunk capacity will be shown later, is defined as a unit time. In the state S_1 , cells arrive at rate ρ_1 , whereas the cell arrival rate is ρ_2 ($\rho_1 > \rho_2$) in the state S_2 . The periods of S_1 and S_2 are assumed to be geometrically distributed, with mean values of T_1 and T_2 unit times, respectively. When $\rho_1=1$ and $\rho_2=0$, it corresponds to the very bursty traffic.

As mentioned in Chapter 2, this model can describe the interactive data traffic and voice call traffic. In the interactive data traffic, the state S_1 corresponds to the active data transfer state, whereas the state S_2 corresponds to the idle period in the data transfer. In a voice telephone call, the state S_1 refers to talk spurt and the state S_2 refers to pause in human voice. For compressed video communication, the frame length corresponding to the state S_1 and the inter-frame periods corresponding to the state S_2 , may also be assumed to be approximately geometrically distributed [15]. ρ_2 is, in general, not zero due to some non-bursty information transfer between frames. Further discussion of the traffic model to the voice and data traffic is included in Section 4.2. For more accurate modeling of video traffic, a discrete-state continuous-time Markov process with batch arrivals may be applied [16].

(a) State Transition of a Single Source at Call and Cell Levels

(b) State Transition of the Number of Sources Connected (at Call Level)

(c) State Transition of the Number of Active Input Lines
when the Number of Connected Sources is y (at cell level)**Figure 3.1 Traffic State Transition Diagrams**

Instead of using $Prob(y/N)$ in equation (2.4), we use $P(y)$ to represent the probability of y calls out of the total nc sources getting connected. Similarly, we have

$$\begin{aligned} P(y) = Prob(y/nc) &= \binom{nc}{y} P_c^y (1 - P_c)^{nc-y} \\ &= \binom{nc}{y} \frac{\left(\frac{T_1}{T_2}\right)^y}{\left(1 + \frac{T_1}{T_2}\right)^{nc}} \end{aligned} \quad (3.3)$$

The expression (2.2), which is the probability of k input lines out of y connected sources becoming active, can be rewritten as

$$P_a(k/y) = \binom{y}{k} P_1^k P_2^{y-k} = \frac{\binom{y}{k} \left(\frac{T_1}{T_2}\right)^k}{\left(1 + \frac{T_1}{T_2}\right)^y} \quad (3.4)$$

where T_1 and T_2 are the active length and idle length respectively.

The probabilities of transition from S_i to S_j in a single source p_{ij} are found to be

$$p_{11} = 1 - \frac{1}{T_1}, \quad p_{12} = \frac{1}{T_1}, \quad p_{21} = \frac{1}{T_2}, \quad \text{and} \quad p_{22} = 1 - \frac{1}{T_2}. \quad (3.5)$$

The state transition diagram for a source at cell level is shown in Figure 3.1(a).

When considering multiple sources connected, for example, given y sources connected, the state transition probability from the number of active input lines x to another number of active input lines x' is derived as the following:

$$\begin{aligned} PB_x^y(x') &= \sum_{j=0}^x \binom{x}{j} p_{12}^j p_{11}^{x-j} \binom{y-x}{i} p_{21}^i p_{22}^{y-x-i} \\ i &= x' - x + j \quad (0 \leq i \leq y - x) \end{aligned} \quad (3.6)$$

If the number of connected sources is y and the number of active input lines out of the connected sources is x , then m cells will arrive to the multiplexer with the conditional probability

$$\begin{aligned} Pr\{m/(x, y)\} &= \sum_{j=0}^x \binom{x}{j} \rho_1^j (1 - \rho_1)^{x-j} \binom{y-x}{i} \rho_2^i (1 - \rho_2)^{y-x-i} \\ i &= m - j \quad (0 \leq i \leq y - x) \end{aligned} \quad (3.7)$$

The probability distribution of the number of arriving cells in a unit time is given by

$$Pr1(m) = \sum_{x=0}^{nc} \sum_{y=x}^{nc} P(y)Pa(x/y)Pr[m/(x, y)] \quad (3.8)$$

Taking the state transition probabilities at call level (3.2) and at cell level (3.6) into account we can obtain the joint probability of m cells arriving in a unit time and m' cells arriving in the next unit time

$$\begin{aligned} P_J(m, m') = & \sum_{x=0}^{nc} \sum_{x'=0}^{nc} \sum_{y=x}^{nc} P(y)Pa(x/y)Pr[m/(x, y)] \\ & \{h_{y,y}PB_x^y(x')Pr[m'/(x', y)] \\ & + h_{y,y+1}PB_x^y(x')Pa(0/1)Pr[m'/(x', y+1)] \\ & + h_{y,y+1}PB_x^y(x')Pa(1/1)Pr[m'/(x'+1, y+1)] \\ & + h_{y,y-1}PB_x^{y-1}(x')\frac{y-x}{y}Pr[m'/(x', y-1)] \\ & + h_{y,y-1}PB_{x-1}^{y-1}(x')\frac{x}{y}Pr[m'/(x', y-1)]\} \quad (3.9) \end{aligned}$$

Then, when m cells arrive in a unit time, the conditional probability of m' cells arriving in the next unit time is

$$Pt_m(m') = \frac{P_J(m, m')}{Pr1(m)} \quad (3.10)$$

which is the state transition probability shown in Figure 3.2.

3.3 Analysis of the Multiplexer Model

The ATM multiplexer model and its operations are similar to the description in the previous chapter, except that the buffer size here takes a general value while the buffer size is equal to trunk capacity in Chapter 2.

The multiplexer system in queueing analysis is often formed by a two-dimensional Markov chain with the state at the end or the beginning of a unit time represented by (m, q) , where m denotes the number of arrival cells in a unit time and q is the queue length as the result of m cells arriving in a unit time. Now, we define a state probability

Figure 3.2 Transition of the Number of Arrival Cells During a Unit Time

that the queue length becomes q as the result of m cell arriving in a unit time as $QL(m, q)$. Summing over disjoint events, we arrive at the following steady state equations:

$$\begin{aligned}
 QL(m', q') &= \sum_{m=0}^{nc} \sum_{q=0}^Q QL(m, q) Pt_m(m') |_{q'=\min(\max(q-M, 0) + m', Q)} \\
 \sum_{m=0}^{nc} \sum_{q=0}^Q QL(m, q) &= 1
 \end{aligned} \tag{3.11}$$

Hence, $QL(m, q)$ can be determined by a numerical solution of the above equations. It is noted that the above equation is similar in shape to the equations obtained in [13]. However, the scenarios covered by the above equations are more than those in [13], since the stochastic process of call connection is taken into account here while it is not considered in [13].

Using the state probabilities $QL(m, q)$, the average cell loss probability is derived through the summation of the probabilities of the queue length overflowing the buffer

with limit size Q , as follows:

$$\begin{aligned}
 P_{LossQ} &= \frac{\text{average number of cells lost in a unit time}}{\text{average number of arrival cells in a unit time}} \\
 &= \frac{\sum_{m=0}^{nc} \sum_{m'=0}^{nc} \sum_{q=0}^Q QL(m, q) Pt_m(m') (\max(q - M, 0) + m' - Q)}{\sum_{m=0}^{nc} \sum_{m'=0}^{nc} \sum_{q=0}^Q QL(m, q) Pt_m(m') (\max(q - M, 0) + m')} .
 \end{aligned} \tag{3.12}$$

where, in the numerator, $\max(q - M, 0) + m' > Q$.

3.4 Simulation and Numerical Results

In this section, we investigate and present our results on the case where nc sources carrying pure bursty traffic are connected. By pure bursty traffic we mean that $\rho_1=1$ and $\rho_2=0$. The reasons for focusing on this case are: (1) to reduce the amount of programming and numerical calculations, (2) to emphasize the worst case study of an ATM multiplexer performance, i.e., relative worst case of average cell loss possibility (with the same number of sources the average cell loss probability of the deterministic connection case is higher than the stochastic connection case; also, in most circumstances, the stochastic connection process is not easy to estimate in advance); (3) idle load usually is much smaller when comparing with active load, and in most cases it is zero: for example, in data traffic and voice traffic cases; (4) examining the effect of active load is equivalent to studying the effect of trunk capacity.

An example of comparing theoretical results and simulation results is presented in Figure 3.3 and Figure 3.4 for relatively short-burst traffic and long-burst traffic, respectively. From the figures, one observes that the theoretical results are confirmed very well by simulations. One can also see that with the increase of trunk capacity, the cell loss probability decreases. Comparing Figure 3.3 and Figure 3.4, we find that to achieve the same cell loss probability, traffic with long bursts will require higher

Figure 3.3 Cell Loss Probability vs Trunk Capacity: $T_1=200$ and $T_2=270$

Figure 3.4 Cell Loss Probability vs Trunk Capacity: $T_1=40771$ and $T_2=50000$

bandwidth than short-burst traffic. In other words, under the same burstiness factor, long burst-traffic will more easily cause congestion than short-burst traffic.

Variations of Buffer Size

Figure 3.5 shows cell loss probabilities against various values of buffer size for a short-burst traffic case. It can be observed that as buffer size is increased, the cell loss probability is slowly decreased.

For the long-burst case cell loss does not decrease significantly with increasing buffer size for size in the range shown in Figure 3.6. It would require buffer size corresponding to the burst length to show a significant decrease.

Comparing with Figure 3.3 and Figure 3.4, we also find that the change of cell loss probability is not sensitive to the change of buffer size, particularly in the long-burst traffic case. To achieve a very small cell loss probability, a very large size of buffer may be needed. However, a large size of buffer may introduce a large cell delay. Therefore, the increased buffer size should be subject to the constraint of cell delay.

Figure 3.5 Cell Loss Probability vs Buffer Size: $T_1=200$ and $T_2=270$

Figure 3.6 Cell Loss Probability vs Buffer Size: $T_1=40771$ and $T_2=50000$

Variations of Burstiness Factor

The average cell loss probability versus trunk capacity with various burstiness factors for a short-burst traffic case, is shown in Figure 3.7, in which $T_1=200$. From the figure, we can see that with the increase in the burstiness factor the trunk capacity required to achieve certain cell loss probability is decreased. Figure 3.8 shows a similar effect for a long-burst traffic case where $T_1=50000$.

Variations of the Number of Sources

Figure 3.9 and Figure 3.10 show cell loss probability against the number of sources. With the increase in the number of connected sources, cell loss probability also increases. From the figures, one can find that cell loss probability is sensitive to the change in the number of sources. Therefore, taking action on the acceptance of a new call and on the control of the number of allowed calls is one of the critical measures to guarantee required cell loss probability.

Figure 3.7 Cell Loss Probability vs Trunk Capacity: $T_1=200$

Figure 3.8 Cell Loss Probability vs Trunk Capacity: $T_1=50000$

Figure 3.9 Cell Loss Probability vs Number of Sources: $T_1=200$ and $T_2=270$

Figure 3.10 Cell Loss Probability vs Number of Sources: $T_1=40771$ and $T_2=50000$

3.5 Conclusions

An approach to obtain average cell loss probability for an ATM multiplexer with a general buffer size has been described in this chapter. The approach involves solving a set of linear equations. The theoretical results are confirmed by simulations.

From our numerical analyses, we conclude that:

1. Enlarging buffer size can reduce cell loss probability. However, an increase in buffer size may not be always feasible, particularly in the constraint of cell delay.
2. Trunk capacity, or the bandwidth allocated, has a crucial effect on the average cell loss probability. To guarantee a small cell loss probability, a large enough bandwidth should be provided.
3. The average cell loss probability is sensitive to the number of connections. Therefore, controlling the number of accepted sources will be effective in ensuring a small cell loss probability.

Chapter 4

Cell Loss Process

4.1 Introduction

The high ratio of propagation to transmission and queueing delay in future B-ISDN networks has made cell loss probability replace average waiting time or throughput as the dominant performance metric. Due to the bursty characteristic of ATM traffic, it has been realized that in order to guarantee a good quality of service (for example, in real-time applications with signal reconstruction), average cell loss probability alone cannot provide enough information for effective traffic control. In ATM networks, a large amount of information loss in a small portion of time will be encountered even if the average cell loss probability can be kept to a very low, satisfactory level. This greatly challenges those bases and concepts which are suitable for conventional network design. In particular, error control based on average cell loss probability is no longer adequate for ATM networks. As an example, Shacham and Kenney [18] observed that loss correlation in a finite-buffer system could completely eliminate the advantage of three orders of magnitude predicted for forward-error correction under the assumption of independent (Bernoulli) losses.

In order to effectively design ATM networks, a complete understanding of the cell loss process is necessary. Unfortunately, there is no exact cell loss model tied to ATM networks which is ready for use. A study of the cell loss process is inevitable.

Recently, a few authors have investigated related aspects of loss process. Kamitake and Suda [17] compute the steady state loss rate for a given number of active callers and then consider the variation in the number of active callers in computing the amount of uninterrupted time from when this loss rate first exceeds a value ζ , until it drops below another value η , with $\eta < \zeta$. Schulzrinne and Kurose [11] characterize the stochastic

properties of *time-out loss periods* for infinite queues, that is, uninterrupted intervals during which the virtual wait is at or above some fixed threshold.

Li [9] studies information loss in packet voice systems due to blocking and clipping and provides a solution for average blocking and non-blocking periods which are different from packet loss and non-loss periods. However, the idea in his approach is useful in finding out a way to tackle the cell loss process. Van Doorn [19] and Meier-Hellstern [20] characterize the overflow process from a finite Markovian queueing system.

However, none of the above-mentioned research provide accurate and complete information about the cell loss process, such as average cell loss period, average non-loss period, average number of cells lost in loss period, loss rate in loss period, and loss period variance for ATM networks. The work in this chapter overcomes the drawbacks of previous analyses and presents theoretical solutions for a set of practical descriptors for the cell loss process. The theoretical results are confirmed by simulations.

4.2 Types of Traffic Covered by Cell-Loss Analysis

In general, it is unrealistic, or at least extremely difficult, to obtain an exact solution of the cell loss process by covering all kinds of traffic. Therefore, in this section, we focus on long-burst traffic which, we believe, will cause severe cell loss problems. Long-burst traffic can be found in large file and image data transfer in ATM networks. Although our analytical solution is intended for long-burst traffic, it will be shown by simulation and numerical analysis that our theoretical approach can provide a good approximation for some not very short-burst traffic, for instance, high threshold voice traffic.

Large File And Image Data Transfer. Consider an ATM multiplexer multiplexing multiple connections with LANs. A typical example is SUN/UNIX environment. We consider bulk data transmission such as large file and image data transfer. A representative traffic characteristic can be found in [21] and [17]. The average data

size is assumed to be 2160 Kbytes (=17280 K bits), which is approximately equal to 40771 ATM cells (each source contributes one cell in a unit time). The resulting probability that a data transfer which is now active becomes idle at the next unit of time is only 2.45×10^{-5} . The probability that more than one source change their state simultaneously at the next unit time is much smaller, which can be neglected. The average idle time between data transfer corresponding to the user mode CPU processing time used for compile/assemble/link is assumed to be 156 seconds (it may be shorter if it is not doing much processing or no processing at all). The LAN speed is 10 Mbits/sec (maybe 10 Mbits/sec~100 Mbits/sec for high speed LAN), thus, the average time for the data transfer is 1.728 sec.

High Threshold Voice Traffic. In most packet voice systems, each voice source is considered to be 64Kbit/s PCM digital signal [9]. Each packet consists of 1 Kbit information which corresponds to 16 ms packetization interval. The values of talkspurt length and idle length can be substantially different based on the sensitivity of the speech activity detection threshold set at the voice source. In particular, a high threshold and a low threshold corresponding to a long voice holding time and short holding time, respectively, are suggested to describe the level of correlation in voice. Two common set of thresholds in literature (e.g., [9], [22], and [23]) are used. That which corresponds to high threshold has the talkspurt length of 1366ms and the idle length of 1802ms. The other, which corresponds to the low threshold, has the talkspurt length of 400ms and the idle length of 600ms. If we convert the above thresholds in terms of cell unit in an ATM transmission case, we can ascertain that for the high threshold, the talkspurt is around 201 ($\frac{1366 \times 1000}{16 \times 53 \times 8}$) cells long and the idle length is around 265 ($\frac{1802 \times 1000}{16 \times 53 \times 8}$) cells long. For the low threshold, the talkspurt is around 58 ($\frac{400 \times 1000}{16 \times 53 \times 8}$) cell long and the idle length is around 88 ($\frac{600 \times 1000}{16 \times 53 \times 8}$) cell long. We are more interested in high threshold voice traffic than low threshold voice traffic, since the former will cause more degradation in cell loss performance than the latter.

4.3 Analysis Of Cell Loss Process

Consider an ATM multiplexer system with nc sources which are all connected and which has a buffer size of Q . The output trunk capacity of the multiplexer is assumed to be M , as defined in previous chapters. The traffic from each source is again considered to be bursty, with active and idle states. The distribution of the active and idle states are assumed to be geometric, with the mean active length T_1 and the mean idle length T_2 .

Similar to the analysis in Chapter 3, the multiplexer system in performance analysis can be formed by a two-dimensional Markov chain, with the state represented by (m, q) where m denotes the number of active sources in the active state and q is the queue length in the buffer. The model is justified by the fact that since slot generation is faster than changes in m by several orders of magnitude, m can be viewed to represent the cell arrival process. The process of m is again characterized by a birth-death process due to the substantially long active and idle lengths.

We will focus on the cell loss process when arrival cells overflow the buffer, rather than the blocking process [9] when the buffer gets full. In fact, cell loss can happen even if the buffer is not full because the number of cells arriving in a time unit may exceed the sum of the number of buffer spaces left over from a previous time unit and the maximum number of cells the trunk can transmit in a time unit.

Denote by (i, Q) the state of the queue length becoming Q as a result of i cells arriving causing cell loss. $\phi_{i, Q} = \phi(i, Q)$ is defined as the equilibrium probability in state (i, Q) .

Obviously, $\phi_{i, Q} = 0$ for $i \leq M$ since the buffer will never be full when the rate of the queue change is negative. Cells are possibly lost only within states (i, Q) for $i > M$. These states are called loss states.

Define

$$\vec{\phi}_Q = (\phi_{M+1, Q}, \phi_{M+2, Q}, \dots, \phi_{nc, Q}) \quad (4.1)$$

as the row vector of equilibrium probabilities in loss states. The sojourn time within these states is called loss period.

Using the results of $QL(m, q)$ and $Pt_m(m')$ obtained in Chapter 3, we derive the following expression for $\phi_{i,Q}$:

$$\phi_{i,Q} = \phi(i, Q) = \sum_{m=0}^{nc} \sum_{f(q)>0}^Q QL(m, q) Pt_m(t) \quad (4.2)$$

where $f(q) = \max(q-M, 0) + i - Q$, and $i \in (M+1, \dots, nc)$.

The cell arrival process is modeled by a continuous-time birth-death process, with the assumption of all sources being connected at the call level. The cell arrival transition rate matrix is denoted by $P = [p_{ij}]$ with the transition rate from i to j given by

$$p_{ij} = \begin{cases} \frac{1}{T_1} & \text{if } j = i - 1 \\ -\frac{1}{T_1} - \frac{nc-1}{T_2} & \text{if } j = i \\ \frac{nc-1}{T_2} & \text{if } j = i + 1 \\ 0 & \text{else} \end{cases} \quad (4.3)$$

for $i, j \in (0, 1, \dots, nc)$. The equilibrium probability of i cells arrival is equal to

$$p_i = \binom{nc}{i} \rho^i (1 - \rho)^{nc-i} \quad \text{for } i = 0, 1, \dots, nc \quad (4.4)$$

where $\rho = \frac{T_1}{T_1 + T_2}$.

The transition rate between loss states is equal to the corresponding cell arrival transition rate, independent of buffer size Q . That is, the transition rate from (i, Q) to (j, Q) is given by p_{ij} , e.g., Fig. 4.1.

It is clear that each loss period is characterized by a transient birth-death process. ϵ_i in Fig. 4.1 is the probability for a loss period to start in state (i, Q) . Let $\vec{\epsilon} = (\epsilon_{M+1}, \epsilon_{M+2}, \dots, \epsilon_{nc})$ with $\vec{\epsilon} \vec{\epsilon}^T = 1$. $\vec{\epsilon}$ is a column unit vector.

By the nature of burst birth-death, a loss period always ends at the transition from the state $(M+1, Q)$ to state (M, Q) . Since there is only one way out of each loss period, the probability distribution of $\vec{\epsilon}$ is independent and identical to each loss period. In other words, the alternation of loss and non-loss periods form an alternating renewal process.

Figure 4.1 Loss state transition diagram

Define t_{ij} for $i, j \in (M+1, \dots, nc)$ as the sojourn time in state (i, Q) during the loss period, provided that the loss period starts in state (i, Q) . Define G_t as the transition rate matrix of this transient birth-death process and T as the matrix of $[t_{ij}]$ with

$$\bar{T} = [\bar{t}_{ij}] \quad (4.5)$$

where $\bar{t}_{ij}$ is the expectation of t_{ij} . From [24], we can get

$$\bar{T} = -G_t^{-1}, \quad (4.6)$$

in which t_{ij} is further shown to be

$$t_{ij} = \sum_{k=M+1}^{\min(i,j)} \frac{1}{p_{k,k-1}} \prod_{l=k}^{j-1} \frac{p_{l,l+1}}{p_{l+1,l}} \quad \text{for } i, j = M+1, \dots, nc. \quad (4.7)$$

Since the equilibrium probability of i cells arrival can also be written as

$$p_i = p_0 \prod_{l=0}^{i-1} \frac{p_{l,l+1}}{p_{l+1,l}} \quad (4.8)$$

we may rewrite t_{ij} as

$$t_{ij} = \sum_{k=M+1}^{\min(i,j)} \frac{1}{p_{k,k-1} p_k} = T_1 \sum_{k=M+1}^{\min(i,j)} \frac{p_1}{k p_k} \quad (4.9)$$

By simple probabilistic arguments we obtain

$$\phi_Q = c \vec{\varepsilon} T, \quad (4.10)$$

or

$$\vec{\varepsilon} = -c^{-1} \phi_Q G_t. \quad (4.11)$$

Here c is introduced for the normalization $\vec{\varepsilon} \vec{\varepsilon} = 1$, given by

$$c = -\phi_Q G_t \vec{\varepsilon}. \quad (4.12)$$

We can show that

$$G_t \vec{\varepsilon} = (-p_{M+1,M}, 0, \dots, 0)^T, \quad (4.13)$$

or

$$c = p_{M+1,M} \phi_{M+1,Q} = \frac{M+1}{T_1} \phi_{M+1,Q}. \quad (4.14)$$

Furthermore, denote by $\bar{t}$ the mean duration of loss period and t_i to be the portion of $\bar{t}$ in state (i, Q) . Accordingly, we have

$$(t_{M+1}, t_{M+2}, \dots, t_{nc}) = \vec{\varepsilon} T. \quad (4.15)$$

From equations (4.6), (4.11), and (4.14), it gives

$$t_i = \frac{T_1}{M+1} \frac{\phi_{i,Q}}{\phi_{M+1,Q}}. \quad (4.16)$$

A special case at $i=M+1$ shows that

$$t_{M+1} = \frac{T_1}{M+1} , \quad (4.17)$$

which is independent of queue and only determined by the link capacity M and the mean bursty length T_1 . By definition

$$\bar{t} = \bar{\epsilon}^T T \bar{\epsilon} , \quad (4.18)$$

hence

$$\bar{t} = \frac{T_1}{M+1} \sum_{i=M+1}^{nc} \frac{\phi_{i,Q}}{\phi_{M+1,Q}} , \quad (4.19)$$

which is the length of the loss period. The second moment of the loss period is expressed from (4.18) as

$$t^2 = \bar{\epsilon}^T \bar{T}^2 \bar{\epsilon} . \quad (4.20)$$

From [24] we can write in a vector-matrix form

$$\bar{T}^2 \bar{\epsilon} = -G_t^{-1} [-2G_t^{-1} - I] \bar{\epsilon} . \quad (4.21)$$

Then using (4.11) yields

$$\begin{aligned} t^2 &= \bar{\epsilon}^T (-G_t^{-1}) [-2G_t^{-1} - I] \bar{\epsilon} \\ &= \frac{T_1}{(M+1)\phi_{M+1,Q}} \phi_Q [-2G_t^{-1} - I] \bar{\epsilon} \\ &= \frac{T_1}{(M+1)\phi_{M+1,Q}} \phi_Q 2\bar{T} \bar{\epsilon} - \bar{t} \\ &= \frac{2T_1}{M+1} \sum_{i=M+1}^{nc} \frac{\phi_{i,Q}}{\phi_{M+1,Q}} \sum_{j=M+1}^{nc} \bar{t}_{ij} - \bar{t} , \end{aligned} \quad (4.22)$$

or

$$t^2 = \frac{T_1}{M+1} \sum_{i=M+1}^{nc} \frac{\phi_{i,Q}}{\phi_{M+1,Q}} \left[2T_1 \sum_{j=M+1}^{nc} \sum_{k=M+1}^{\min(i,j)} \frac{p_j}{kp_k} - 1 \right] . \quad (4.23)$$

The variance of loss period is equal to

$$\sigma_t^2 = \bar{t}^2 - t^2 . \quad (4.24)$$

Now let us look at the non-loss period defined as s which is very important since it can tell us how often a loss period occurs. Since the system is ergodic with overall loss probability equal to

$$\sum_{i=M+1}^{nc} \phi_{i,Q} , \quad (4.25)$$

it means

$$\frac{\bar{t}}{\bar{t} + s} = \sum_{i=M+1}^{nc} \phi_{i,Q} . \quad (4.26)$$

Solving $\bar{s}$ from the above equation gives

$$\bar{s} = \frac{T_1 \left(1 - \sum_{i=M+1}^{nc} \phi_{i,Q} \right)}{(M+1)\phi_{M+1,Q}} = \frac{T_1}{(M+1)\phi_{M+1,Q}} - \bar{t} . \quad (4.27)$$

The average cell loss rate in the entire loss period is

$$\begin{aligned} r_{ll} &= \frac{\text{Average Number of Cells Lost in Loss Period}}{\text{Average Number of Cells Arriving in Loss Period}} \\ &= \frac{\sum_{i=M+1}^{nc} \bar{l}(i) \bar{t}_i}{\sum_{i=M+1}^{nc} i \bar{t}_i} \\ &= \frac{\sum_{i=M+1}^{nc} \bar{l}(i) \phi_{i,Q}}{\sum_{i=M+1}^{nc} i \phi_{i,Q}} . \end{aligned} \quad (4.28)$$

where $\bar{l}(i)$ is the average number of cells lost when in state (i, Q) , i. e.,

$$l(i) = \frac{\sum_{m=0}^{nc} \sum_{f(q)>0}^Q [\max(q-M, 0) + i - Q] QL(m, q)}{\sum_{m=0}^{nc} \sum_{f(q)>0}^Q QL(m, q)} . \quad (4.29)$$

in which $f(q) = \max(q-M, 0) + i - Q$.

The average number of cells lost in the loss period is thus obtained

$$l_n = \sum_{i=M+1}^{nc} \bar{l}(i) \bar{t}_i = \sum_{i=M+1}^{nc} \frac{\bar{l}(i) T_1 \phi_{i,Q}}{(M+1) \phi_{M+1,Q}} . \quad (4.30)$$

We can assume that the lost cells are most evenly distributed among all the sources which are currently active. Then, all the above cell loss process measures also represent the measures with respect to each individual source.

It is found that the above results are directly expressed by the loss state probability vector $\vec{\phi}_Q$ and the other parameters such as T_1 , T_2 , nc , and M which are the system parameters. Noticing (4.2) we also observe that $\vec{\phi}_Q$ depends on $QL(m, q)$ which can be solved using the approach presented in Chapter 3. Therefore, all of the above results finally dependent on $QL(m, q)$.

4.4 An Object-Oriented Cell Loss Simulator

This section describes the motivation for and the development of an object-oriented simulation environment for the class of ATM problems considered in this thesis. A reader who wish to concentrate on numerical results at a first reading could omit this section and resume with section 4.5.

C programming language has been proved to be efficient for software development, for example, in simulation software construction [25]. As a part of our research, a special-purpose simulation of an ATM multiplexer aiming at evaluating its average performance, in particular, the average cell loss probability, was created. It is experimentally shown that simulation in C is ten times faster than using the simulation package QNAP2 (if 10 seconds are used by C simulation, 100 seconds will be used by QNAP2).

However, with advances in research, the software now needs to be enhanced or even rewritten in order to include new aspects of the study and to support new functions. This will lead to a tremendous amount of work. For instance, if more types of traffic are considered and the cell loss process needs to be provided, the whole software will have to be reexamined in order to implement the correct codes in the right places. If the software is enhanced by later users, even more difficulties and frustrations will be expected. The documentation and maintenance of the software will also be difficult.

After a careful examination of the software design options available, I decide to use an object-oriented approach to build my simulation software. With an object-oriented design, software enhancement becomes easier. We need only to design and bring in new objects, avoiding the overall change.

Considerations of system portability, computer resource requirements, developmental and run-time efficiency, support of numerical computation, and so forth, made me decide in favor of a C++ implementation over other implementations of object-oriented programming. C++ is a very clean, object-oriented extension to C [26]. It is relatively mature, having been in use inside AT&T for quite a while. It has no special run-time support requirements, aside from the presence of the standard C libraries. Best of all, it is a candidate for worldwide acceptance, just like Unix and C before it.

In this section, some basic concepts and ideas of object-oriented programming as they are used in my simulation software design, are first introduced. A conceptual design and implementation of ATM multiplexer simulation software (ATMSS) to study average system performance and cell loss process is then described.

4.4.1 Object-Oriented Programming Preliminary

Object-oriented programming is a paradigm in which a software system is developed to consist of many subsystems based on objects. Operation is done by objects exchanging information among themselves. This paradigm improves software reusability, extensibility and maintainability. It also provides a natural method of software development.

which is similar to the way in which humans approach problems in nature.

Traditional software systems are usually decomposed into subprograms or modules based on their functions which are identified by algorithmic abstractions. In traditional programming languages such as FORTRAN or the old version of PASCAL, programs are often written around a main routine which calls on other subroutines to function at appropriate times. It is up to the main program to make decisions. The subroutines can only play a subordinate role. In contrast, object-oriented programming allows every object to make its own decisions by function association. One of the main advantages of object-oriented programming is that program upgrades can be done by introducing new objects. This greatly simplifies software development and evolvability.

Object. Object is an entity which includes its own data and function to manipulate the data called method. Only the function owned by the object can access its data. The initialized values of the data of an object will persist indefinitely throughout its life-time unless varied by some functions. Thus, the data collectively represent the state of the object. In other words, only the functions of an object can change its state. An object, as a whole, can be viewed as a dynamic, self-functional entity, and the internal details of its implementation are shielded from the user of the object. Users can use an object without understanding the details. This abstraction allows the developer to extend a software system by adding in new definitions without modifying earlier ones. In C++ [27], with "friend" declaration, objects can interact with each other. With higher levels of control to perform message passing, objects can also communicate with each other.

Class. In C++ object-oriented programming, objects are usually not defined individually. Instead, a class definition provides a template for generating any number of objects, each one an identical copy of a basic prototype. Classes are basically extensions to the structure of C. With classes, one can create user-defined types that not only represent data, but the operations to be carried out on the data. Classes are built into two parts: (1) a class declaration and (2) a class implementation. A class declaration lists all the members that belong to the class, including its data members and member functions. The collection of member functions is called the class implementation.

Inheritance. Although classes are mutually distinct, they do share some fundamental properties. C++ provides a way to define and extend new classes economically, that is, class inheritance. With this feature, a class definition can specify from which classes the new class will be inherited, i.e., automatically obtain all their definitions. One does not need to copy or rewrite such common data and functions. Thus, consistency in definitions is retained and confusion is avoided.

A special case of class inheritance is that a class inherits only from one other class called its parent. The classes form a tree structure with a root class as the most general class. The root class generally provides definitions for functions needed to interface with higher level procedures. A more specialized class may override such general definitions by providing functions of its own with the same name. Object-oriented programming systems can do the dynamic binding—interpreting the meaning of a function in its proper context. This important feature distinguishes message passing from conventional function calls. Object-oriented systems may evolve by adding more and more specialized classes, while the higher level procedures need not be changed so long as the newly-introduced specialized functions are compatible (same input and output). A more general form of class inheritance is when a class inherits from several classes.

Object-oriented design. Object-oriented design is a kind of software design methodology based on objects. It usually involves the following steps:

1. Use logic and pseudo-code to describe the major functions to be performed in the software systems.
2. Identify the objects that are part of a solution to the problem. Define the member data that these objects will need.
3. Identify operations to be taken on or by the objects to effect a solution to the problem.
4. Abstract and group the possible objects into classes, according to their properties. Develop a new hierarchy of new classes required for a solution.
5. Complete objects' final implementation.

The above steps need not be followed in strict order in software development. Some of them can be embedded into a single, integrated step.

4.4.2 Object-Oriented Design of ATM Multiplexer Simulation Software (ATMSS)

Design Objectives

Extension, reuse, and maintenance of software objects are the main purposes we sought from our adoption of an object-oriented programming technique. Program reusability is a goal of great promise. Simply being organized will not help one design a reusable software. To achieve software reusability, one needs to identify object behaviors that are useful in more than one context. Designing objects to be independent of others will make software extension and maintenance easy. We do not intend to construct yet another general-purpose simulation language; nor do we attempt to build a general-purpose performance simulator to be used for all situations.

The first objective is to make simulation software customized for individual research questions. A research simulation model should contain no more detail than is required to explore the particular question at hand. The strategy is to design reusable software objects. With the objects, it is expected that the major simulation design work would be in choosing and interconnecting objects of interest, and linking those objects with one's own approach related to the problem.

The second objective is that the simulation software should be easily modified and that extended and some parts of it could be reused. Since we do not know what future research questions might be, we can only be certain that new functionalities will, eventually, need to be added. The economy of effort in programming can be achieved by proper definition and implementation of basic classes so that the possible inheritance of objects is expected.

Finally, the overall objective is to make the software easily understood, that is, both the individual objects and the way they work together.

Design Guidelines

The design principles which are deducted from the objectives are as follows.

Classes should be as simple as possible, meaning that each class performs only a small set of major functions. For example, the traffic class contains only its own characteristics. The number of different classes is increased in order to keep individual objects simple.

The relationships among objects should be kept natural, simple, consistent, logical, and unambiguous. The interactions between objects should not rely on the internal data structure of objects. For example, the classes Traffic, Source, ATM_mplxer, and Timer are identified by their corresponding physical entities, and their relationships inherit their natural ones.

4.4.3 Conceptual Design and Implementation

Involving Object-Oriented Concepts

An object-oriented model of an ATM multiplexer system consists of objects like traffic, source, and ATM multiplexer. The state variables of these objects change in response to events, such as the arrival of a cell or the departure of a cell; these events happen at discrete points in time. There is a natural one-to-one correspondence between physical objects in an ATM multiplexer system and instances of software objects that represent them. The encapsulation within software objects of member data and functions for updating the state variables provides an integrate software modularity.

The inheritance of functions and instance variables by use of a hierarchical class structure permits the addition of complexity and functionality to simple objects, as necessary, by creating subclasses of existing classes. Other objects that work with the original simple versions will still work with the new complex objects. This feature is very useful in constructing special-purpose simulations for research purposes because it avoids unnecessary complexity, permitting run time efficiency, and also avoids confusion.

The Conceptual Framework

In the following, the words representing objects that have corresponding classes in the simulation system are underlined.

In discrete event simulation, the time sequence of real-world events is reproduced by the simulation; the state of the simulated system changes only at the discrete times when events occur. The simulation clock (Timer) is advanced at a discrete period of time.

An ATM multiplexer system is composed of Traffic, Source, multiplexer (ATM_mplxer) and so on. Traffic generated from each source is independent of each other. The incoming traffic is statistically multiplexed into the buffer and then transmitted on the output trunk. Here, we are focusing on the traffic with bursty nature, which is considered to be the prominent characteristic of traffic in future BISDN networks. For some particular traffic, such as motion video and video phone, we only need to modify or inherit the class Traffic without an overall software change.

Traffic. Since a two-state bursty traffic model is employed, the traffic is characterized by two instance variables, i.e., the mean burst length and the mean idle length. The burst descriptor of traffic, the burst factor, is evaluated by the function **Burst_factor**.

Source. This involves an object hierarchy design. The priority is placed on the functional specifications of individual objects, instead of defining any inheritance hierarchy to start with. With the interest in homogeneous traffic sources, one can identify that each source generates a similar type of traffic. Every source has common instance variables *mean_burst* and *mean_idle*, and a common function **Burst_factor**. This gave rise to the subclass **Source** of the superclass **Traffic**. The state variables of **Source** are the random variables *burst_length* and *idle_length*, which are assumed to be geometrically distributed in our research and generated by the function **assign_bi_length**. The functions **dec_burst_length** and **dec_idle_length** decrement the burst length and the idle length respectively. The burst length and the idle length at a particular point in time are reported by functions **fburst_l** and **fidle_l**.

ATM_mplxer. An ATM multiplexer **ATM_mplxer** can be modeled to be primarily composed of buffer and trunk. Its state variables are *buffer_size*, queue length (*q_length*), and trunk capacity (*trunk_bility*), defined previously. The function **inc_q_length** increments queue length whenever a cell arrives. If queue length is larger than the value of trunk capacity at the end of each unit time, the function **update_q_length** will reduce the queue length by the amount of trunk capacity. However, if the queue length is smaller than the value of the trunk capacity at the end of each unit time, the function **reset_q_length** will clear the buffer. The function **adj_q_length** will adjust the queue length to the buffer size when the queue length overflows the buffer. The function **fq_length** reports the value of the queue length for performance evaluation. The class **ATM_mplxer** allows the other classes **Timer** and **Perform_ave** and the function **act_proc** to access its data and methods for data collection and manipulation purposes by a way of conveniencing object interactions.

Timer. The class **Timer** is used to control the simulation clock, simulation transient length, and the total simulation period. The instance variables are maximum simulation time (*max_time*) in time unit, *current_time*, total simulation time (*sim_time*), and transient period *tran_time*. The functions **new_time** and **total_sim_time** will increment the current simulation clock and find the total simulation time at the end of simulation. The class **Perform_ave** and the function **act_proc** are permitted to interact with the class **Timer**.

Perform_ave. The class **Perform_ave** is a software object used to evaluate average system performance, particularly the average cell loss performance. Its data members are naturally those performance variables, i.e., *num_arr* (the number of cell arrivals), *num_drop*, *num_arri* (the number of cell arrivals in an unit time), *num_dropi* (the number of cells dropped in an unit time), *num_dropi_pre* (the number of cells dropped in the previous unit time), throughput *Thr*, mean number of cells in the system *L*, system utilization *U*, mean resident time *W*, cell loss probability *P_loss*, and two middle results *B* and *S*. The functions **new_num_arr** and **new_num_drop** increment the counter of cell arrivals and the counter of cells dropped, respectively. In a unit

time, the number of cell arrivals is increased by function **new_num_arri**; while the number of dropped cells is updated by **new_num_dropi**. At the end of each unit time, instance variables *num_dropi_pre*, *num_arri*, and *num_dropi* are reset by function **reset_num_i**. The average performance variables *Thr*, *L*, *U*, *W* and *P_loss* are calculated by function **find_perform**. The functions **l_BS** and **s_BS** are employed to obtain the middle results *B* and *S*. The functions **fnum_dropi_pre**, **fnum_dropi**, **fnum_arri**, **f_num_drop**, **f_num_arr**, **f_Thr**, **f_U**, **f_L**, **f_W**, and **f_P_loss** will return the values of *num_dropi_pre*, *num_dropi*, *num_arri*, *num_drop*, *num_arr*, *Thr*, *U*, *L*, *W* and *P_loss* respectively.

Perform_proc. The software object **Perform_pro** is actually a cell loss process data collector. Its data member set is constructed by the instance variables describing the cell loss process, i.e., *cnt_nlperiod* (counter of non-loss period), *cnt_lperiod* (counter of loss period), *arr_inloss* (cell arrivals in loss period), *sumlperiod* (sum of non-loss period), *sumnlperiod* (sum of non-loss period), *sum_lpi* (loss length variable), *sum_lp2* (sum of the square of loss length), *ave_lperiod* (average loss period), and *ave_nlperiod* (average non-loss period). The counter of loss period is incremented by the function **new_cnt_lperiod**. When a non-loss period is first encountered, the function **set_enter_nlperiod** increments the counter of non-loss period, calculates *sum_lp2*, and resets *sum_lpi* to zero. When in loss period, the variables *arr_inloss*, *sumlperiod*, and *sum_lpi* are updated by the function **set_items_inloss**. The variable *sumnlperiod* is incremented by function **new_sumnlperiod**. The functions **find_avelperiod** and **find_avenlperiod** find and return the average loss period and the average non-loss period respectively. The function **Ploss_inloss** calculates the loss rate in the loss period. The average number of cells lost in the loss period is found by **aveno_lossinloss**. The function **var_lperiod** provides the variance of loss period.

Fig. 4.2 shows the interactions among objects. The diagram is simply a network in which the nodes represent objects and the arcs, along with arrows, represent accessibility of one object to another object. For example, **Perform_proc** is accessible to **Perform_ave**.

Figure 4.2 Fig. 1 Object Interaction Diagram

4.4.4 Discussion of Object-Oriented ATMSS Design

I feel that I have been reasonably able to achieve the design objectives set at the beginning of the design. During the ATM simulation software development, I tried to adhere to the design guidelines as closely as possible. Effectiveness, or the degree of reusability, is determined mainly by anticipating which objects are likely to change. The strategy to make these objects as simple and independent as possible (for example, **Traffic**) is useful in localizing and reducing the impact of changes. If one wants to study other traffic control issues with simulation, one can build one's own objects and plug into the ATMSS. It is noted that for ease of future extension, the simulator in the ATMSS is not separated as an object for the time being.

In the design, the objects were identified based on their correspondence to physical objects and important conceptual performance measurement events in simulation. When the design was started, it was found that lack of established design guidelines, and absence of documentation of successful case studies, was a clear shortcoming in pursuing object-oriented design. But as experience with the design process continued, it was no

longer perceived as a problem.

When beginning to use object-oriented programming techniques, more time and effort are required to become comfortable with the design principles and the documentation of the software objects. However, once this is done, modification and extension of the objects becomes easier than writing code in conventional programming languages.

The software productivity gains can be achieved using object-oriented programming. In the design stage, the gains come from two aspects. First, real-world objects and concepts are directly and intuitively mapped to software objects. On the other hand, algorithm specifications and data structures are much more difficult to derive using the conventional software development approach. Second, the conceptual design is less dependent on the details of implementation than in conventional approaches.

It should be pointed out that this object-oriented ATM simulation software is an extension to the conventionally-designed one. It is obvious that the ATMSS avoids using external variables. This reveals one aspect achieved in this software: easier extensibility and reusability. An extensive test run and partial comparison with the results of the conventionally-designed ATMSS show the correctness of the object oriented ATMSS.

4.5 Simulation Confirmations

In this section, simulations and numerical computations are carried out to verify our analytical results.

4.5.1 long-burst traffic

Simulations and numerical calculations are carried out to confirm our theoretical results. For example, consider an ATM multiplexer system with the fixed parameters $T_1=50000$, $T_2=100000$, the number of connections $nc=12$, and the buffer size $Q=15$. With the varying of trunk capacity, we obtain a group of figures showing cell loss process descriptors against trunk capacity.

Fig. 4.3 is the cell loss period versus trunk capacity. With the increase in trunk

Figure 4.3 Cell Loss Period Versus Trunk Capacity

capacity, cell loss period decreases.

Fig. 4.4 shows cell non-loss period versus trunk capacity. It is observed that an increase in trunk capacity leads to an increase in cell non-loss period.

Fig. 4.5 indicates the average number of cells lost in the entire average cell loss period. With the increase in trunk capacity, the average number of cells lost in the loss period reduces.

Fig. 4.6 shows the variance of cell loss period against trunk capacity.

Fig. 4.7 is the loss rate in the loss period versus trunk capacity. It is clear that the loss rate in the loss period is larger than the average loss rate in the entire operation period. With the increase in trunk capacity, the loss rate decreases.

4.5.2 High Threshold Voice Traffic

The following example gives a flavor of a good approximation of applying our

Figure 4.4 Cell Non-Loss Period Versus Trunk Capacity

Figure 4.5 Average Number of Cells Lost In Loss Period

Figure 4.6 The Variance Of Cell Loss Period Versus Trunk Capacity

Figure 4.7 Cell Loss Rate In Loss Period Versus Trunk Capacity

analytical approach to short burst traffic. A group of high threshold voice traffic is examined. In the following study, we fix all other system parameters except the trunk capacity M .

A set of parameter values are selected for simplicity of the analyses. The number of connected voice sources are assumed to be $nc=20$. The buffer size is also $Q=20$. The simulation and numerical results for the high threshold case (talkspurt length $T_1=200$ and idle length $T_2=265$) with trunk capacity $M=8$, $M=9$, $M=11$, are shown in Table 4.1. The trunk capacity represents the portion of the total cell bandwidth allocated to the voice sources. For instance, if a voice source has a 64kbps speed, the allocated bandwidths are correspondingly 512kbps, 576kbps, and 704kbps.

From Table 4.1, one can find that the relative discrepancy of analytical results from simulations is around 5%, not excluding the simulation confidence interval.

4.5.3 Low Threshold Voice Traffic

To show the discrepancy of applying our method to even shorter burst traffic, a group of low threshold voice traffic is analyzed. In the following study, we also fix all other system parameters except the trunk capacity M . The number of connected voice sources are assumed to be $nc=20$. The buffer size is also $Q=20$. The simulation and numerical results for the low threshold voice case (talkspurt length $T_1=58$ and idle length $T_2=88$) with trunk capacity $M=8$ and $M=10$ are shown in Table 4.2.

From Table 4.1, one can find that the relative discrepancy of analytical results from simulations is around 13%, not excluding the simulation confidence interval.

4.6 Numerical Results

In this section, we use numerical method to examine how system parameters affect cell loss performance.

**Table 4.1 Comparison of Simulations and Analytical
Results for the High Threshold Voice Case**

nc=20	M=8		M=9		M=11	
	Analytical Results	Simulation Results	Analytical Results	Simulation Results	Analytical Results	Simulation Results
Loss Period	76.946	73.607	53.048	55.376	31.944	33.432
Non-loss Period	82.554	79.000	118.519	123.677	360.383	376.483
Loss period variance	13024.429	12342.017	5857.125	6183.024	1757.070	1872.564
Loss Rate	0.136	0.136	0.076	0.076	0.016	0.016
Loss Rate in Loss Period	0.232	0.229	0.187	0.190	0.131	0.134

4.6.1 Variations of Buffer Size

Here, as an example, we consider only long-burst traffic. The traffic characteristics are selected to be $T_1=50000$ and $T_2=354000$. The value of T_2 corresponds to one second idle time.

Table 4.3 and 4.4 show cell loss behaviors against various buffer sizes for 12 and 18 sources, respectively. From the tables, one can see that cell loss performance is not very sensitive to the change in buffer size. However, one interesting phenomenon can be observed from the tables: that with the increase in buffer size, other cell loss

**Table 4.2 Comparison of Simulations and Analytical
Results for the Low Threshold Voice Case**

nc=20	M=8		M=10	
	Analytical Results	Simulation Results	Analytical Results	Simulation Results
Loss Period	20.352	23.063	12.316	13.920
Non-loss Period	42.520	47.973	137.495	154.231
Loss Period Variance	769.723	901.204	224.683	266.878
Loss Rate	0.094	0.094	0.020	0.020
Loss Rate in Loss Period	0.215	0.223	0.151	0.158

performance indexes have been improved, except for the average cell loss period and the number of cells lost in loss period which, on the contrary, show a minute increase.

The effect of buffer size suggests that it may not always be feasible to enlarge buffer size in order to obtain better cell loss performance. Consider certain real time traffic, for example, voice [9] and image, which, at certain times, may tolerate a small amount of cell loss. Although a large buffer size can make cell loss happen less frequently because of an increased non-loss period, the increased average loss period and the number of cells lost in loss period may lead to an unexpected distortion of received voice and picture.

4.6.2 Variations of Burstiness Factor

To examine the effect of the burstiness factor on cell loss performance, we fix the length of active period T_1 to be 50000 and vary the length of idle period T_2 . Two

Table 4.3 Cell Loss Performance vs Buffer Size at M=8 and nc=12

nc=12	Q=20	Q=40	Q=60	Q=80	Q=120
Loss Period	5798	5798	5800	5800	5802
Non-loss Period	5.521×10^{10}	5.540×10^{10}	5.558×10^{10}	5.577×10^{10}	5.615×10^{10}
Loss period variance	3.614×10^8	3.615×10^8	3.616×10^8	3.617×10^8	3.618×10^8
Loss Rate	7.373×10^{-7}	7.350×10^{-7}	7.328×10^{-7}	7.305×10^{-7}	7.260×10^{-7}
Loss Rate in Loss Period	0.115	0.115	0.115	0.115	0.115
No. of Cell Loss in Loss Period	6046	6047	6049	6050	6053

particular values of T_2 , that is, $T_2=354000$ corresponding to a short user processing time of one second, and $T_2=1770000$ corresponding to a longer user processing time of five seconds, are investigated.

Figure 4.8 shows the average cell loss probability with the change in the burstiness factor, with $T_1=50000$ being fixed against trunk capacity. It is clear that with the increase in the burstiness factor, the average cell loss probability decreases.

Table 4.4 Cell Loss Performance vs Buffer Size at M=8 and nc=18

nc=18	Q=30	Q=40	Q=60	Q=80	Q=120
Loss Period	6343	6345	6347	6349	6352
Non-loss Period	5.526×10^8	5.552×10^8	5.552×10^8	5.569×10^8	5.586×10^8
Loss period variance	4.980×10^8	4.982×10^8	4.984×10^8	4.987×10^8	4.990×10^8
Loss Rate	5.862×10^{-5}	5.855×10^{-5}	5.841×10^{-5}	5.827×10^{-5}	5.814×10^{-5}
Loss Rate in Loss Period	0.125	0.125	0.125	0.125	0.125
No. of Cells Lost in Loss Period	7217	7219	7225	7230	7235

The average cell loss period corresponding to two different burstiness factors versus trunk capacity, is given in Figure 4.9. The larger the burstiness factor, the shorter the average cell loss period.

The change in the average non-loss period with a varied burstiness factor against a varied trunk capacity can be found in Figure 4.10. With the increase in the burstiness factor the average non-loss period also increases.

Figure 4.11 is the average loss rate in a loss period for two different burstiness factors. The average loss rate in a loss period is reduced when the burstiness factor becomes large.

The number of cells lost in a loss period with two different burstiness factors versus trunk capacity is presented in Figure 4.12. The increase in burstiness factor leads to a decrease in the number of cells lost in a loss period.

Figure 4.8 Average Loss Rate vs Trunk Capacity

4.6.3 Variations of Number of Sources

Again, to analyze the effect of the number of sources, we fix the average active length to be $T_1=50000$.

Figure 4.13 presents an average cell loss rate with various numbers of sources. The increase in the number of sources leads to the increase in average cell loss rate.

The average cell loss period against the number of sources for different burstiness

Figure 4.9 Average Loss Period vs Trunk Capacity

factors is shown in Figure 4.14. Average cell loss period becomes larger when the number of sources increases.

The change in the average non-loss period with a varied number of sources can be seen in Figure 4.15. In contrast to the increase in the average cell loss period, the average non-loss period decreases with the increase in the number of sources.

Figure 4.16 gives the average cell loss rate in a loss period with the change in the number of sources. The average cell loss rate in a loss period also increases with the increase in the number of sources. It is noted that the average cell loss rate with a high burstiness factor increases slowly when the number of sources increases.

The average number of cells lost in a loss period versus the number of sources, is found in Figure 4.17. As the number of sources increases, the average number of cells lost in the loss period increases as well. However, for a high burstiness factor,

Figure 4.10 Non-loss Period vs Trunk Capacity

the average number of cells lost in the loss period increases slowly with the increase in the number of sources.

4.6.4 Effect of Traffic Arrival Rate

One of the most promising traffic control schemes in ATM networks is traffic shaping. Basically, traffic shaping slows down the arrival traffic into the networks, that is, to change the cell arrival rate.

The examination of the traffic arrival rate is equivalent to the investigation of trunk capacity, which has been defined as the maximum number of cells carried by trunk in a unit of time. If the traffic arrival rate becomes low, the unit of time for generating one cell from each source will be longer and, therefore, within this longer unit time the

Figure 4.11 Loss Rate in Loss Period vs Trunk Capacity

trunk can carry more cells. Instead of varying the traffic arrival rate, we can change the trunk capacity to look at the resulting cell loss performance.

Tables 4.5 and 4.6 show cell loss behavior against various trunk capacities for 30 and 40 sources, respectively. In the tables, a trunk capacity increase corresponds to a decrease in the traffic arrival rate. From the tables, one can observe that the trunk capacity or the traffic arrival rate has a great effect on the cell loss performance. With the increase in trunk capacity or the decrease in the traffic arrival rate, the cell loss behaviors become much better. In general, applying traffic shaping function to the incoming traffic can significantly improve the cell loss performance. However, the cost and the difficulties of providing a large buffer in network intermediate nodes would suitably push the traffic shaping function to be performed to the edge of the networks.

Figure 4.12 Number of Cells Lost in Loss Period vs Trunk Capacity

4.6.5 Examination of Consecutive Loss

The cell loss behavior results obtained so far have revealed one unique and important characteristic of cell loss patterns corresponding to bursty traffic: consecutive loss. This correlated cell loss phenomenon has made error control, based on the retransmission of individual cells in conventional data packet switching networks, inefficient and inadequate for high speed networks.

An error control feature, based on the retransmission of a block of cells seems to be the best candidate for ATM networks to cope with consecutive loss. We will discuss this error control feature further in the next chapter.

To design an effective error recovery scheme, especially the block size, one has to make use of cell loss information. In particular, the information concerning how many

21 of/de 2

PM-1 3 1/2" x 4" PHOTOGRAPHIC MICROCOPY TARGET
NBS 1010a ANSI/ISO #2 EQUIVALENT

PRECISIONSM RESOLUTION TARGETS

PIONEERS IN METHYLENE BLUE TESTING SINCE 1974

1800 COUNTRY ROAD & BURNINGWELL AVE. DEEP SEA
TEL: 912 408 7887 FAX: 912 408 7887 TLA 048888888

Figure 4.13 Average Cell Loss Rate vs the Number of Sources

cells from each source, on average, are lost when cell loss starts to happen. If we have this statistic—the average number of cells lost in a loss period for each source, we can reasonably select it as the block size. All cells lost will naturally be acknowledged and retransmitted as one block.

As shown in the above analyses, we have been able to obtain cell loss behavior for all sources, particularly the average number in a cells lost in loss period. Since the identical traffic model for all sources is employed in the analysis, we can reasonably assume that all sources have the same kind of cell loss behavior. Therefore, the average number of cells lost in a loss period from each source is equal to the average number of cells lost in a loss period from all sources, divided by the number of sources.

Tables 4.7 and 4.8 show cell loss behavior against various numbers of sources and trunk capacities when networks provide around 10^{-9} and 10^{-6} cell loss probabilities,

Figure 4.14 Average Cell Loss Period vs the Number of Sources

respectively. In the tables, the average burst length is selected to be $T_1=50000$ and the average idle length is $T_2=354000$. The reason for choosing these long bursty lengths and short idle lengths as our investigated example is to design a robust error recovery scheme based on the real-world observed relative worst case. This is because one second of user processing time, which corresponds to an idle length of 354000, is quite short in a UNIX/SUN environment.

From our numerical analyses, as shown in Table 4.7, we have found that $nc=11$ and $M=9$ are the minimum number of sources and trunk capacity to provide a 10^{-9} cell loss probability. Accordingly, the average number of cells lost in a loss period from each source can be calculated from Table 4.7 as $\frac{5128.434}{11} \approx 466$ cells. When the network operates with its maximum capacity, that is, $M=15$, corresponding to 15 times of high-speed local area network's capacity of 10Mbps, the number of sources which it can

Figure 4.15 Average Non-loss Period vs the Number of Sources

connect to provide a 10^{-9} cell loss probability is 26. The associated average number of cells lost in a loss period from each source is $\frac{3705.637}{26} \approx 143$ cells. Therefore, the block size can be designed to contain 140~470 cells, depending on the number of frequent connections. Regardless of the number of sources, and for the sake of robust function, the block size can be selected to be 470 (cells).

Similarly, from Table 4.8, one can identify that the block size should be chosen within 121~365 when the network operates at 10^{-6} cell loss probability.

From our numerical results, it seems that the lower the average cell loss probability the network can provide, the larger the block size. However, cell loss will happen less frequently when the network operates at a low cell loss probability.

Figure 4.16 Average Loss Rate in Loss Period vs the Number of Sources

4.7 Discussion And Concluding Remarks

This chapter has presented an approach to study the cell loss process. The theoretical results are confirmed by simulation.

It is found in simulation that with a larger trunk capacity value, and hence a smaller average cell loss probability, a longer simulation run is needed in order to obtain accurate cell loss process descriptors. This is similar to the situation when we tried to use simulation to obtain a much smaller cell loss probability.

From the simulations and the numerical results given in this chapter, we can conclude that the analytical approach can give accurate results for voice traffic, particularly with high threshold values.

Figure 4.17 Average Number of Cells Lost in Loss Period vs the Number of Sources

Similarly, we can conclude that our analytical approach can provide an accurate approximation for the medium size (in the order of several hundred cells) file transfer in ATM networks.

Cell loss performance is not sensitive to the change in buffer size. Large buffer size does not necessarily lead to better cell loss behavior because of the increased loss period and the number of cells lost in the loss period. The number of sources and the trunk capacity have a noticeable effect on cell loss performance. Admission control and bandwidth allocation must consider resultant cell loss behavior. Traffic shaping is effective in improving cell loss performance. For error control in ATM networks, block acknowledgment and retransmission is more efficient for consecutive loss. A block size at the order of several hundred cells is suggested.

Table 4.5 Cell Loss Performance vs Trunk Capacity: Buffer Size=nc=30

nc=30	M=3	M=6	M=12
Loss Period	31292	11448	4623
Non-loss Period	29099	153612	1.906×10^9
Loss period variance	2.166×10^{10}	2.389×10^9	2.885×10^8
Loss Rate	0.295052	0.028835	7.816×10^{-7}
Loss Rate in Loss Period	0.413	0.205	0.091
No. of Cells Lost in Loss Period	66155	17670	5531

Table 4.6 Cell Loss Performance vs Trunk Capacity: Buffer Size=nc=40

nc=40	M=7	M=14
Loss Period	11565	4253
Non-loss Period	89988	9.261×10^8
Loss period variance	2.817×10^9	2.670×10^8
Loss Rate	0.040	1.175×10^{-5}
Loss Rate in Loss Period	0.200	0.083
No. of Cells Lost in Loss Period	20187	5389

Although we only present examples with the number of traffic sources not being very large, our analytical approach does not impose any limit on the number of sources, nor on the trunk capacity. However, with the increase of the number of sources, more

Table 4.7 Cell Loss Performance vs Number of Sources and Trunk Capacity when Networks Operate at the Level of 10^{-9} Cell Loss Rate

Number of Sources and Trunk Capacity	nc=Q=26 M=15	nc=Q=17 M=12	nc=Q=11 M=9
Loss Period	3405	4006	5064
Non-loss Period	7.301×10^{12}	1.717×10^{13}	6.155×10^{12}
Loss period variance	1.348×10^8	1.725×10^8	2.622×10^8
Loss Rate	1.578×10^{-9}	1.154×10^{-9}	6.120×10^{-9}
Loss Rate in Loss Period	0.068	0.080	0.101
No. of Cells Lost in Loss Period	3706	4170	5128

Table 4.8 Cell Loss Performance vs Number of Sources and Trunk Capacity when Networks Operate at the Level of 10^{-6} Cell Loss Rate

Number of Sources and Trunk Capacity	nc=Q=39 M=15	nc=Q=26 M=12	nc=Q=16 M=9
Loss Period	3848	4415	5409
Non-loss Period	5.746×10^9	1.293×10^{10}	1.638×10^{10}
Loss period variance	2.083×10^8	2.456×10^8	3.342×10^8
Loss Rate	1.700×10^{-6}	1.215×10^{-6}	1.800×10^{-6}
Loss Rate in Loss Period	0.076	0.087	0.107
No. of Cells Lost in Loss Period	4716	5279	5843

physical memory is required to numerically obtain the result. This is because the number of sources, together with the buffer size, contribute to the dimension of a matrix used in our numerical calculations. With the increase in the dimension, the time needed to solve the matrix is also increased. However, compared with simulation, the numerical solution time is much shorter. On average, the numerical solution time is only one percent of the simulation time. In our experience, the numerical solutions are obtainable for 40 sources and a buffer size of 40 cells, or 12 sources with 150 cells of buffer size in a Spar II workstation with 30MB memory.

Chapter 5

Evaluation of Traffic Control in ATM Networks and a Proposal

5.1 Introduction

Traffic control functions in ATM networks are carried out at three levels, the call process level [28] (where congestion can prevent the establishment of a new call), the packet level [29] (where congestion can cause the delay or loss of packets as opposed to cells), and the ATM level [30] (where congestion can cause the delay or loss of cells).

At the call process level, virtual channels are set up before the cell transfer phase begins for each call. New connections must be admitted to the network in a manner which protects the network performance for all existing connections; new connections must be withheld from portions of the network in a congested or nearly congested state. At the packet level, action must be taken to prevent an excessive packet flow on any packet-carrying connection, and in some cases provide features which help portions of the network recover from excessive packet flows. At the ATM level, cells must be ordered for processing by ATM concentrators and ATM switches, action must be taken to prevent an excessive cell flow on any connection, and in some cases it is appropriate to provide functions which help portions of the network recover from excessive cell flows.

In general, traffic control in ATM networks falls into two major categories: congestion control and error control. Congestion occurs when the demand is greater than the available resources. In conventional packet switching networks, flow control is embedded into error control (X.25). This strategy is no longer suitable for high speed ATM networks since it will lead to service congestion. In ATM networks, the bottleneck problem has been shifted to the processing speed at the network switching nodes and the propagation delay of the channel, because the high-speed channel increases the ratio of propagation delay to cell transmission time, and the ratio of processing time to cell

transmission time. It is difficult to apply hop-by-hop control schemes. Therefore, flow control in ATM networks is cooperating with congestion control.

Figure 5.1 ATM Traffic Control Model

A suitable set of congestion controls for a broadband network will include features for admission control, buffer and queue management, traffic policing or enforcement and reactive control. Their execution points are shown in Fig.5.1. The purposes of admission control are to establish fair blocking and to ensure that the resources available to each admitted connection are sufficient to meet performance objectives for cell loss ratio, cell transfer delay and cell delay variation. The purpose of buffer and queue management is to order cells for the ATM level, efficiently processing in order to satisfy various performance requirements. The purpose of traffic policing is to monitor a connection's resource usage for compliance with appropriate limits, and to act on violations of those limits. The purpose of reactive control is to relieve congestion once it has occurred in some portions of the network.

In this chapter, we will discuss various traffic control features from the point of view of cell loss performance and other performance measures.

5.2 Various Traffic Congestion Controls and Associated Problems

Congestion control is a challenge for a broadband network. Most traffic sources in ATM network supports are bursty. Bursty traffic can cause severe network congestion. In a high-speed network environment, a very short period of congestion can lead to a large amount of information lost, and can therefore badly degrade the quality of service. Although high speed channels provide efficient multi-services, they also increase the difficulties in traffic management and limit the applicable congestion control mechanisms. The simplicity of congestion control protocols is highly expected in an ATM network in order to eliminate a possible bottleneck in protocol processing time. It is necessary to push most of the link-by-link layer protocols to higher edge-to-edge layers. Besides the significant modification of conventional reactive control, preventive control has been proposed for an ATM network. In the following discussion, several congestion controls, that is, admission control, buffer and queueing management, traffic policing and stop-and-go queueing, fall into the class of preventive control.

5.2.1 Admission Control

Admission control can be performed at the call process and the ATM levels. At the call process level, it is necessary that the use of high bandwidth services does not consume available resources to such an extent that lower bandwidth services experience an unacceptable increase in blocking probability. At the ATM level, new connections should be admitted to the network only if sufficient network resources are available to support that connection in conformance with the relevant performance objectives. For a

connectionless type of VBR service, an underlying connection is assumed to have been established through subscription.

The simplest rule for the allocation of bandwidth for each new call is to reserve the peak bit rate, regardless of the burstiness. However, if the presence of bursty traffic is high, the network resources are not optimally utilized. One of the main purposes of the ATM is to achieve a high resource utilization; otherwise, other transfer modes (e.g., STM) can be preferred, as they are simpler and more cost-effective. Therefore, it is better to allocate a bandwidth less than peak bit rate in order to take advantage of statistical multiplexing. However, this will lead to an unavoidable cell loss when the buffers are full. In this case, an accurate admission rule is mandatory to keep the cell loss probability small, as allowed by the QOS requirement.

Most admission control research focuses on the ATM level because of the consideration of congestion avoidance (e.g. [7], [17], [31, 32]). Since the maximum permitted load in any ATM network link depends on the specific performance and traffic characteristic of individual connections, the admission control criterion is more appropriately defined by differentiating connections to be accepted into multiple performance and traffic classes [33, 34]. A class of performance is identified by connections with the same performance requirements in terms of delay and loss rates. A class of traffic is identified by connections with the same statistical properties. Two kinds of criteria are used to determine whether to accept a new connection. One is based on the equivalent bandwidth measure W [Mbits/s]. The other, suggested in the literature, is based on the cell transmission delays and the cell loss probabilities. From the results obtained in this thesis, the author believes that the second criteria, based on the cell transmission delays and cell loss behavior, will be more appropriate.

In the case where heterogeneous traffic sources (a general case of homogeneous sources) are multiplexed, an equivalent bandwidth criterion for a new connection (i), obtained recently in [31, 32], can be generalized as

$$W_i = f(L_i, N_{ji}, P_{ji}, \gamma_{ji}) \leq C - W_1 - \dots - W_{i-1}, \quad (5.1)$$

where the subscript i indicates the i th performance class, j means the j th traffic class, N_{ji} is the number of connections of the j th class, γ_{ji} is the average traffic density, P_{ji} is a burstiness indicator, C is the channel capacity, L_i is the permitted load of the i th class, and $W_1, \dots$ and W_{i-1} are the equivalent bandwidths of $i-1$ connected classes. If condition (5.1) is not satisfied, the new connection required is not allowed. The advantage of this approach is that it is not limited to a small size of network, as is the one in [34]. However, the problem found in this approach is that the specific performance requirements, such as cell loss ratio and cell transmission delay constraints, are not related and the highly bursty traffic case cannot be accurately handled.

As for the admission decision made by the performance requirements, the approaches using long-term, time-averaged values in the past (e.g., [35, 36]) are found to be insufficient in an ATM network, due to the rapidly and dynamically varying traffic [7, 17]. Networks can be temporarily congested when the traffic is highly bursty and when it changes dynamically, and a large amount of information will be lost during congestion periods, even when the long-term, time-averaged value of loss rate is kept small. Recently, an instantaneous cell loss probability was used as a decision criterion to consider the temporal behavior of a network [37, 17]. A new connection is accepted if the instantaneous cell loss rate is kept below a threshold value at each switching node for longer than a predetermined percentage of time. However, the implementation of this approach may not be practical in terms of the complexity of calculating the instantaneous cell loss probability. Furthermore, since the period of high instantaneous cell loss is very short, the utilization efficiency of channel capacity will be low if high bandwidth is assigned for the entire transmission time, according to the requirement of instantaneous cell loss. We believe that bandwidth allocation and admission decisions based on cell loss performance are more realistic and accurate. This admission strategy can control network performance and can guarantee a more precise cell loss requirement by users. The implementation of this strategy is not difficult. For example, for a new connection request, one first evaluates the resulting loss performance and then decides whether or not to accept the connection, based on the obtained cell loss information.

This will, at the first level, guarantee cell loss performance in the networks. It would be a good idea to shape the source traffic in order to provide a satisfactory cell loss performance when accepting new connections. In other words, terminal adaptors or source communication adaptors should be placed at the transport network accesses to shape input traffic profiles by means of appropriate buffering.

5.2.2 Buffer and Queueing Management

In the management of buffers and queues of ATM cells, the ordering of cells for different classes of traffic for processing at ATM concentrators and ATM switches has a significant influence on the network efficiency achievable with various mixes of CBR and VBR traffic, and on the quality of service. In general, multilevel priorities are necessary to achieve effective statistical multiplexing. There are two ways of using priorities [7]. One way is to use priority as a scheduling method which schedules delay-sensitive traffic for service first. The other way is to use priority as a local congestion control scheme which selectively discards low priority cells when network congestion occurs.

Priority Scheduling Method. A static priority scheme, which always schedules the high priority traffic first, is found not to be suitable for an ATM network since the performance for the low priority traffic will be severely degraded if a large portion of the network traffic consists of high priority traffic. A dynamic priority scheme is more favorable. Among many dynamic priority schemes, Queue Length Threshold (QLT) and Head-of-the-Line with Priority Jumps (HOL_PJ) [38] are the most promising ones. In the QLT scheme, priority is given to the loss-sensitive traffic when the number of loss-sensitive cells in the queue exceeds some threshold value; otherwise, priority is given to the delay-sensitive traffic. This scheme only differentiates between delay-sensitive and loss-sensitive data in order to meet the performance requirements of each traffic type. If different delay requirements exist within the delay-sensitive traffic, the HOL_PJ scheme must be used. In the HOL_PJ scheme, higher priority is given to the class of traffic with stricter delay requirements. Each priority class forms its own queue, and cells within the same priority class are served FCFS. A limit is imposed on the maximum

queueing delay of cells within each queue: when the waiting time of a cell exceeds the maximum delay limit, that cell jumps to the end of the next higher priority queue. The performance of different classes can be controlled by adjusting the values of the delay limit. However, there are still some questions not answered on the processing overhead.

Priority Congestion Control Scheme. In this scheme, priority is given to loss-sensitive traffic (for example, data) over loss-insensitive traffic (for example, voice). When congestion occurs, the cells from lower priority classes are discarded first. The different cell loss requirements of different classes of traffic are identified by this priority scheme. In discarding cell information, an improvement can be obtained by selectively discarding cells containing less important information. For instance, the voice quality can be maintained by discarding cells carrying background noise [39].

Recently, a space priority mechanism has been suggested [40, 41] for congestion control in order to satisfy cell loss requirements. In this mechanism, cells are classified into two classes: Class_1 and Class_2. Cells are stored in a queue of finite size, and served FIFO. As soon as the buffer is full, arriving Class_2 cells are discarded, while an arriving Class_1 cell is allowed to enter if it can take the place of a waiting Class_2 cell, which is then discarded. A Class_1 cell is lost only if no Class_2 cell is waiting. One problem with this mechanism is the complicated buffer management, since the cell sequence has to be preserved in an ATM environment even though using selective cell discarding is used. Therefore, further improvement for this mechanism is needed.

5.2.3 Traffic Policing

Once a connection has been admitted to the network, it is necessary to ensure that the usage of network resources by such a connection does not exceed limits established at the time of its admission. If this is not done, insufficient network resources may be available to provide the desired quality of service on other connections. This function is of paramount importance, especially if a statistical multiplexing approach is applied. Such functions must act on each source before all traffic is multiplexed, otherwise, it may be very difficult to identify the cause of an incorrect behavior. The corrective

actions performed by the policing unit still act on all the multiplexed users, the honest ones included.

The Leaky Bucket (LB) mechanism is the first and best understood policing algorithm in the literature [42]. The other variations, such as Killing, Jumping and Moving window (or Exponential Weighted Moving Average) have almost comparable performances. LB is based on an estimation of the average bandwidth actually used by a source. When actual bandwidth exceeds a chosen threshold, new incoming cells are dropped until the average monitored value goes down to the threshold value. However, because of the statistical variation of the cells' generation, the control parameters must be set with a large tolerance in order to guarantee the contracted mean rate; thus, the policing function is ineffectual. On the contrary, if the chosen policing parameters are too close to the declared mean rate, only small infrequent variations are admitted and a high cell-dropping rate is observed. In this case, the policing is unacceptable and downright dangerous.

A modified version of the LB mechanism, the Virtual Leaky Bucket (VLB), has been studied and proposed [34] in order to overcome the limitations of the existing policing method. Unlike the LB approach, the VLB scheme considers the policing together with the congestion control problem. As opposed to the LB mechanism, the VLB scheme does not drop those violated cells at the network interface, but marks them. All cells enter the network and are dropped by a network node only if a congestion situation occurs.

In the implementation of the VLB mechanism, the marking function is performed by setting a violating tag in the cell header, and the network nodes must manage the buffer with priority, using a threshold or overwriting scheme. A possible implementation of the overwriting scheme, using a status table and shift registers, is proposed in [43]. Another way of implementing the overwriting scheme is by using memory accessed by pointers and organized in a list. In the list, the marked cells' elimination is the same as an ordinary record elimination from the list and does not involve shifting the stored records.

The main advantage of VLB over LB is that the excess cells generated by the traffic sources are dropped only if necessary, that is, when node congestion occurs. When this

is the case, the dropping of cells is first limited to marked cells; only when this traffic reduction is not enough are some unmarked cells dropped.

The VLB mechanism seems to have become increasingly attractive nowadays. However, a very important problem has been largely ignored: when traffic sources are very bursty, the VLB mechanism may frequently cause network congestion, and therefore more cell loss since a large number of violated cells are tagged and allowed to enter the network. With VLB, the performance requirements of highly bursty traffic can hardly be met.

One way to eliminate this problem is by using traffic shaping, as previously mentioned. The purpose of traffic shaping is to throttle cell inputs into a network to avoid the bursty cell transmissions. The shaping function can be executed by admission control either at user-network interface or at a data source by buffering and injecting cells into the network at a slower speed. Although our results obtained in previous chapters have supported the traffic shaping mechanism as an effective measure to improve cell loss performance, it may, at the same time, slow down the transmission of traffic. However, if it is applied to delay tolerable, highly bursty traffic, and if the VLB mechanism acts on delay-sensitive and loss-insensitive highly bursty traffic, the network performance might be greatly improved.

5.2.4 Stop-and-Go Queueing

A congestion control mechanism called Stop-and-Go queueing (SG_Q), aiming at providing congestion-free communications with guaranteed throughput and constant end-to-end delay per connection, has been proposed in [44]. This strategy is composed of two parts: an admission policy imposed per connection at the source node, and a particular queueing discipline performed at the switching nodes. The admission policy requires the packet stream of each connection to possess a certain smoothness property upon arrival at the network, while the queueing discipline preserves this property as packets travel inside the network.

Admission Policy. After a transmission rate r_k is allocated to a connection k , its packet arrival to the network is required to be (r_k, T) -smooth. During each frame of length T , once the source node (the network node to which a user is connected) receives $r_k T$ packets from connection k , no more packets are admitted until the next frame starts. Alternatively, the allocated r_k is required to be large enough so that the stream of packets arriving at the network always maintain the (r_k, T) -smoothness.

Stop-and-Go Queueing. The above admission policy guarantees that the traffic stream of each connection k is (r_k, T) -smooth. If this property continues to hold as the packet stream of each connection arrives to the intermediate switching nodes, then there will be no congestion problem. However, this is not the case. As packets of a connection proceed to the destination node, they tend to cluster together and form longer and longer bursts which will violate the smoothness property. The SG_Q is an alternative to the conventional FIFO queueing that completely guarantees the original smoothness. In SG_Q, transmission of a packet which has arrived at any link during a frame should always be postponed until the beginning of the next frame. The implementation of the SG_Q is relatively simple [44] since it is actually a delayed-FIFO queueing.

However, this framing strategy will incur two performance costs: the extra delay of the stop-and-go queueing, and the strict admission policy of policing smoothness property on packet arrivals. In particular, if the framing strategy is uniformly applied to all of the services in a broadband network, low utilization of the transmission capacity will be inevitable. Therefore, in my view, this strategy is only suitable for periodic bit streams, moderately bursty service, and services with stringent packet loss, but not for delay-sensitive traffic.

5.2.5 Reactive Control

In a heavily loaded ATM network, localized congestion can still occur even with effective mechanisms for call admission, buffer and queue management, and traffic policing. The statistical nature of VBR traffic could produce temporary overloads at some internal network buffers. In such a case, certain additional congestion control

procedures should be employed to help the congested portions of a broadband network recover to their normal state. This local congestion control function can be realized by reactive control. However, none of the conventional link-to-link reactive controls can be effectively applied to the high-speed network because of the significantly slow feedback. Some researchers are negative about implementing reactive control mechanisms in ATM networks [17]. I believe that reactive control is necessary in an ATM network's local node and that it would be very effective if it is applied at the right time, when the local congestion meter reaches a certain threshold.

In order to reduce the distance feedback information propagation, more effective reactive control should be performed between network users and the edge of the network, as in [45]. Alternatives for such reactive controls include declining to accept new connections based on such measures as equipment utilization, buffer fill and alternate routing.

5.3 Error Control and Recovery Schemes

Although the high-speed transmission media underlying the ATM layer in a broadband network have provided a very low channel error rate, errors, particularly errors due to cell loss, can still occur. To guarantee an error-free transmission, error control and recovery schemes are indispensable. Due to the different characteristics of ATM networks from conventional packet switching networks, one cannot indiscriminately apply those error control and recovery schemes in packet switching networks to ATM networks. For example, window flow control in the form of coupled flow and error control is effectively performed in conventional packet switching networks. But if it is applied to ATM networks, a large window will be required to achieve high throughput since the ratio of propagation delay to cell transmission time is larger in the high-speed network. However, a large window has little control effect, because an entire window may be transmitted at one time, possibly leading to network congestion. Therefore, error control in ATM networks must be independent of flow control. Moreover, another major reason that the error control schemes of conventional packet networks cannot be

directly used in ATM networks is that packet switching uses complex protocols. The overhead entailed by these protocols is substantial. This will increase the processing time in ATM networks and aggravate the performance bottleneck.

The implementation of error control schemes can be carried out in three ways: 1) link-to-link, 2) end-to-end, and 3) a combined link/end scheme. On a link-to-link basis, retransmission of errored or lost cells takes place only between adjacent switching nodes. On an end-to-end basis, retransmission takes place only between sender and receiver. In a combined link/end scheme, both end-to-end and link-to-link retransmissions are allowed. Since a link-to-link scheme involves heavy protocol processing at each switching node, it is not appropriate in the high speed ATM network, and neither is a combined link/end scheme. To reduce the overhead of protocol processing, error control has to be pushed up to the higher end-to-end level. Instead of relying on sliding window protocols implemented at each link, erroneous packets are retransmitted only at the end-to-end level. The feasibility of this end-to-end error control scheme is also supported by the very high-quality, low error-rate transmission media available, since errors occurring on a link are decreased significantly when compared to the case of conventional networks. It has been shown (for example, [46–48]) that the end-to-end scheme performs better than the link-to-link scheme and the combined link/end scheme, requiring fewer network resources.

There are two classes of retransmission strategies: go-back-n and selective-repeat protocols, which can possibly act as the backbones of an error control scheme in an ATM network. In these two strategies, the senders transmit cells continuously without waiting for an acknowledgment from the receiver. Upon receipt of a negative acknowledgment or when time-out occurs, a go-back-n protocol has to retransmit all the cells starting with the lost cell, whereas a selective-repeat protocol only needs to retransmit the lost cells. In ATM networks where the channel speed is very high and the ratio of propagation delay to cell transmission time is very large, the number of cells in transit can be very large. If go-back-n protocol is used, it can cause a significant reduction in throughput since a large number of cell retransmissions are required. The selective-repeat protocol achieves

better throughputs than the go-back-n protocol, since it retransmits only those cells that are negatively acknowledged or whose time-outs have expired. However, the buffers required for reordering cells become large. With the advancement of VLSI technology, more cost-effective reordering buffers will be available. Therefore, the selective-repeat strategy is superior to the go-back-n strategy.

An improvement of selective-repeat protocols is the use of a block (that is, a group of cells) acknowledgment scheme, which is suggested in other networks [49]. Selective-repeat is performed on a block basis, and either block needs to be retransmitted or, only the erred/lost cells needs to be retransmitted. The block acknowledgment schemes reduce the large overhead incurred by sending a separate acknowledgment for each cell, and also reduce the number of bits required to address the erred/lost cell if the entire block is retransmitted. Therefore, the processing time in ATM networks is significantly reduced.

However, it should be noted that block size has a non-negligible effect on the performance of an ATM network. A very large block size will cause traffic load to be increased, especially when traffic is highly bursty, and will therefore reduce throughput. I believe that a block-based, selective-repeat protocol performed on an end-to-end basis, with a proper design of block size, is the most appropriate error control scheme in ATM networks. From the results obtained in the previous chapter, one can conclude that a block size at the order of hundreds of cells should be used to obtain an efficient error recovery gain. Although the block size is derived from the investigation of long-burst traffic, it is also adequate for short-burst traffic since short-burst traffic will, in general, lead to a shorter loss period and a smaller average number of cells lost in loss period.

5.4 A Multi-Paradigm Approach to Control of Integrated Traffic

Summing up the above analyses and discussions, we present here a mutually compatible set of solutions that cover each of the six traffic control features, in order to improve traffic control in ATM networks and to support statistical multiplexing.

5.4.1 Admission Control

This consists of the following two stages:

Stage 1: separating bandwidth pools and dynamically varying the size of bandwidth pools in order to increase the achievable network utilization.

Stage 2: dynamic bandwidth allocation within pools. For CBR connections, sufficient bandwidths are allocated to meet their peak rates. For delay-insensitive bursty traffic sources, traffic shaping is done first in order to obtain an expected good cell loss performance and then less than the peak rates are allocated, or a greater number of connections are accepted. For delay-sensitive bursty connections, bandwidths are allocated to meet their cell loss performance requirements.

5.4.2 Buffer and Queueing Management

Different priority methods are suggested to apply to different traffic with different performance requirements. For delay-sensitive traffic, a priority scheduling method such as HOL_PJ is applied. For loss-sensitive traffic, a priority congestion control method is employed. For traffic with stringent delay and cell loss requirements, the priority scheduling method and the priority congestion control method should be executed together.

5.4.3 Traffic Policing

For delay-insensitive highly bursty traffic, the traffic shaping is performed. For delay-sensitive and loss-insensitive highly bursty traffic, the VLB mechanism can be applied.

5.4.4 Stop-and-Go Queueing

Similar to traffic shaping, SG_Q is best applied to delay-insensitive traffic.

5.4.5 Reactive Control

Reactive control should be executed between network users and the edge of the network.

5.4.6 Error Control and Recovery Schemes

Error control and recovery schemes should be based on an end-to-end basis. A block-based selective-repeat protocol, with a block size at the order of hundreds of cells, should be used as the backbone of an error control scheme.

5.5 Concluding Remarks

This chapter reviews and discusses traffic control problems in ATM networks. A comprehensive approach to meet the traffic control challenge is presented. The fundamental criterion for comparing various traffic control techniques is based on the network utilization, subject to the constraints imposed by end-to-end performance requirements, especially cell loss performance and acceptable implementation costs. A multi-paradigm approach to the control of integrated traffic has been suggested, which provides features for admission control, buffer and queue management, traffic policing, stop-and-go queueing, reactive control, and an error control and recovery scheme. The implementation of this improvement remains unaddressed. However, in my opinion, the complexity of implementing this improvement is acceptable.

Chapter 6

Conclusions

6.1 Conclusions

This thesis has established an ATM multiplexer queueing model incorporated with bursty traffic, and has studied its average cell loss probabilities and cell loss behavior. Traffic control in future high-speed ATM networks, based on cell loss performance, other performance requirements, and the quality of service, has also been evaluated. The main contributions of this thesis are as follows:

- Analytical solutions of average cell loss probabilities for a simple and extended ATM multiplexer queueing model are obtained. The effect of system parameters on average cell loss probability is examined.
- A theoretical approach to analyze cell loss behavior is presented. Analytical solutions of average cell loss period, non-loss period, loss rate in loss period and loss period variance are derived. The effect of system parameters on cell loss performance is also investigated.
- An ATM multiplexer simulation software and, especially, an object-oriented cell loss simulator, are developed to confirm the theoretical results.
- A reasonable improvement of traffic control schemes is suggested through a comprehensive evaluation.

From the analyses in this thesis, the following conclusions are made.

- A low traffic load and connection load, smaller number of connections, and a large burstiness factor can lead to a small cell loss probability.
- A small cell loss probability does not necessarily mean a good cell loss performance. Cell loss performance should accurately be addressed by other variables, in particular, average loss period, non-loss period, and the average number of cells lost in loss period.

- Cell loss behavior is not very sensitive to the change of buffer size when the buffer size is not very large. A large buffer size can reduce cell loss probability, but cannot improve overall cell loss performance. However, a large buffer size will increase the period of cell loss.
- Accepting a smaller number of sources or allocating more bandwidth can improve cell loss performance.
- Cell transmission delay and cell loss behavior, instead of cell loss probability, should be taken as another criterion for admission control.
- Traffic shaping is one of the most effective schemes to improve cell loss performance.
- A block-based error recovery mechanism, with a block size of hundreds of cells, should be used in ATM networks.

6.2 Suggestions for Further Study

Chapter 5 contains a number of proposals for further study of the control of integrated traffic.

The research in this thesis is mainly for data and voice application. Extension to video traffic is needed.

Due to limited physical memory, it will be difficult to obtain a numerical solution of cell loss performance for very large sources. An efficient and fast algorithm is expected to be developed to reduce the matrix dimensionality problem. This is possible since the matrix encountered in the solution is sparse and a decomposition is possible. Neuts' method [50] may be helpful in this pursuit.

References

- [1] R. Vickers, "Broadband ISDN Tutorial #3," *IEEE INFOCOM*, 1990.
- [2] A. P. W. R. Bynrne and M. N. Ransom, "Word-wide standardization of broadband ISDN," *Int. J. of Digital and Analog Cabled Systems*, pp. 182-192, 1988.
- [3] S. E. Minzer, "Broadband ISDN and asynchronous transfer mode (ATM)," *IEEE Communications Mag.*, pp. 17-25, 1989.
- [4] R. Handel, "Evolution of ISDN towards broadband ISDN," *IEEE Network*, pp. 7-13, 1989.
- [5] A. Hac and H. B. Mutlu, "Synchronous optical network and broadband ISDN protocols," *Computer*, vol. 22, no. 11, pp. 26-34, Nov. 1989.
- [6] CCITT, *Draft recommendation I.121-broadband aspects of ISDN*, CCITT, TD49(PLEN), Seoul, Korea, Feb. 1988.
- [7] J. J. Bae and T. Suda, "Survey of traffic control schemes and protocols in ATM networks," *Proc. of the IEEE*, vol. 79, no. 2, pp. 170-189, Feb. 1991.
- [8] R. Kositpaiboon, G. Woodruff, and P. Richard, "ATM traffic control for guaranteed performance," *Int. J. Digital and Analog Commun. Systems*, vol. 3, pp. 117-125, 1990.
- [9] S. Q. Li, "Study of information loss in packet voice systems," *IEEE Trans. Commun.*, vol. 37, no. 11, pp. 1192-1202, Nov. 1989.
- [10] A. B. Bondi, "The influence of cell loss patterns and overheads on retransmission choices in broadband ISDN," *GLOBECOM*, pp. 0444-0447, 1991.
- [11] H. Schulzrinne and J. F. Kurose, "Distribution of the loss period for some queues in continuous and discrete time," *IEEE INFOCOM*, pp. 1446-1455, 1991.
- [12] A. Chandan, "Performance evaluation of an ATM multiplexer," Master's thesis, Dept. of Electrical Engineering, University of Ottawa, Feb. 1990.
- [13] M. Hirano and N. Watanabe, "Traffic characteristics and a congestion control scheme for an ATM network," *Int. J. Digital and Analog Commun. Systems*, vol. 3, pp. 211-217, 1990.
- [14] T. C. Hou and A. K. Wong, "Queueing analysis for ATM switching of mixed continuous-bit-rate and bursty traffic," *Proc. IEEE ICC*, pp. 318.6.1-318.6.8, 1990.
- [15] B. S. Kraimeche, "A channel access structure for wide-band ISDN," *IEEE J. on Sel. Areas in Commun.*, vol. SAC-5, pp. 1327-1335, 1987.
- [16] Y. Yasuda, H. Yasuda, N. Ohta, and F. Kishino, "Packet video transmission through ATM networks," *Proc. IEEE GLOBECOM*, pp. 25.1.1-25.1.5, 1989.
- [17] T. Kamitake and T. Suda, "Evaluation of an admission control scheme for an ATM network considering fluctuations in cell loss rate," *GLOBECOM*, pp. 1774-1780, 1989.
- [18] N. Shacham and P. McKenney, "Packet recovery in high-speed networks using coding and buffer management," *IEEE INFOCOM*, pp. 124-131, 1990.
- [19] E. A. V. Doorn, "On the overflow process from a finite Markovian queue,"

- Performance Evaluation*, vol. 4, pp. 233–240, 1984.
- [20] K. S. Meier-Hellstern, "The analysis of a queue arising in overflow models," *IEEE Transactions on Communications*, vol. 37, pp. 367–372, 1989.
 - [21] E. D. Lazowska, "File access performance of diskless workstations," *ACM Trans. Comput. Syst.*, vol. 4, pp. 238–268, 1986.
 - [22] S.-Q. Li, "A new performance measurement for voice transmission in burst and packet switching," *IEEE Trans. Commun.*, vol. COM-35, pp. 1083–1094, Oct. 1987.
 - [23] J. N. Daigle and J. D. Langford, "Models for analysis of packet voice communication systems," *IEEE Select. Areas Commun.*, vol. SAC-4, pp. 847–855, 1986.
 - [24] R. A. Howard, *Dynamic Probabilistic Systems, Volume 1: Markov Models*. New York: Wiley, 1971.
 - [25] M. H. MacDougall, *Simulating Computer Systems: Techniques and Tools*. The MIT Press, 1987.
 - [26] R. Winder, *Developing C++ Software*. 1991.
 - [27] C. H. Pappas and W. H. Murray, *TURBO C++ Professional Handbook*. Borland-Osborne/McGraw Hill, 1990.
 - [28] M. Decina and T. Toniatti, "On bandwidth allocation to bursty virtual connections in ATM networks," *ICC*, pp. 318.6.1–318.6.8, 1990.
 - [29] J. Y. Hui, "Resource allocation for broadband networks," *IEEE J. Selected Areas in Commun.*, vol. 6, pp. 1598–1608, 1988.
 - [30] K. Bala, I. Cidon, and K. Sohraby, "Congestion control for high speed packet switched networks," *INFOCOM*, pp. 520–526, 1990.
 - [31] A. Lombardo, S. Palazzo, and D. Panno, "Admission control over mixed traffic in ATM networks," *Int. J. Digital and Analog Commun. Systems*, vol. 3, pp. 155–159, 1990.
 - [32] H. Esaki, K. Iwamura, and T. Kodama, "A simple and effective admission control method for an ATM network," *GLOBECOM*, pp. 300.5.1–300.5.6, 1990.
 - [33] G. M. Woodruff, R. G. H. Rogers, and P. S. Richards, "A congestion control framework for high-speed integrated packetized transport," *GLOBECOM*, pp. 7.1.1–7.1.5, 1988.
 - [34] G. Gallassi, G. Rigolio, and L. Fratta, "ATM: bandwidth assignment and bandwidth enforcement policies," *GLOBECOM*, pp. 49.6.1–49.6.6, 1989.
 - [35] L. Dittmann and S. B. Jacobsen, "Statistical multiplexing of identical bursty sources in an ATM network," *GLOBECOM*, pp. 39.6.1–39.6.5, 1988.
 - [36] M. Hirano and N. Watanabe, "Characteristics of a cell multiplexer for bursty ATM traffic," *ICC*, pp. 13.2.1–13.2.5, 1989.
 - [37] 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*, pp. 49.5.1–49.5.7, 1989.

- [38] Y. Lin and J. Kobza, "Analysis of a delay-dependent priority discipline in a multiclass traffic packet switching node," *INFOCOM*, pp. 9A.4.1–9A.4.10, 1988.
- [39] N. Yin, S. Q. Li, and T. E. Stern, "Data performance in an integrated packet voice/data system using voice congestion control," *GLOBECOM*, pp. 16.4.1–16.4.5, 1988.
- [40] G. Hebuterne and A. Gravey, "A space priority queueing mechanism for multiplexing ATM channels," *Computer Networks and ISDN Systems*, vol. 20, pp. 37–43, 1990.
- [41] G. Gravey and G. Hebuterne, "Space priority mechanism: modeling and resolution," *CNET/IAA Technical Report*, 1989.
- [42] J. S. Turner, "New directions in communications (or which way to the information age?)," *IEEE Commun. Mag.*, vol. 25, pp. 8–15, 1986.
- [43] J. S. Turner, "Buffer management systems," *WUCS Internal Report*, 1988.
- [44] S. J. Golestani, "Congestion-free communication in broadband packet network," *ICC*, pp. 308.3.1–308.3.6, 1990.
- [45] A. Gersht and K. J. Lee, "A congestion control framework for ATM networks," *INFOCOM*, pp. 701–710, 1990.
- [46] A. Bhargava, J. F. Kurose, D. Towsley, and G. Vanleemput, "Performance comparison of error control schemes in high-speed computer communication networks," *IEEE J. Selected Areas Commun.*, vol. 6, pp. 1565–1575, 1988.
- [47] T. Suda and N. Watanabe, "Evaluation of error recovery schemes for a high-speed packet switched network: link-by-link versus edge-to-edge schemes," *INFOCOM*, pp. 722–731, 1988.
- [48] M. E. Zarki and N. Shroff, "Performance analysis of packet loss recovery schemes in interconnected LAN-WAN-LAN networks," *High Speed Networking*, vol. III, pp. 285–308, 1991.
- [49] D. D. Clark, M. L. Lambert, and L. Zhang, "NETBLT: a high throughput transport protocol," *Proc. ACM SIGCOMM*, pp. 353–359, 1987.
- [50] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Baltimore, MD: The Johns Hopkins University Press, 1981.
- [51] H. Ichikawa, M. Aoki, and T. Uchiyama, "High-speed packet switching systems for multimedia communications," *IEEE J. Selec. Areas Commun.*, vol. SAC-5, pp. 1336–1345, Oct. 1987.
- [52] D. S. Lee, S. Q. Li, and K. H. Tzou, "Analysis of video packet loss control in ATM networks," *GLOBECOM*, pp. 505.3.1–505.3.5, Nov. 1990.

END

24 05 93

FIN