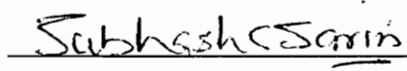


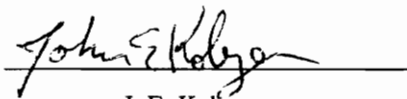
A STUDY AND ANALYSIS
OF
A TRANSMISSION SCHEDULING AND DISCARD ALGORITHM
FOR
ATM NETWORKS
by
Tzu-En Lee

Thesis submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree
MASTER OF SCIENCE
in
Industrial and Systems Engineering

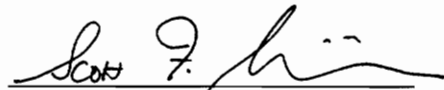
APPROVED:



S. C. Sarin, Chairman



J. E. Kobza



S. F. Midkiff

September 22, 1995

C.2

LD
5655
V855
1995
L46
2

A STUDY AND ANALYSIS
OF
A TRANSMISSION SCHEDULING AND DISCARD ALGORITHM
FOR
ATM NETWORKS

by

Tzu-En Lee

Dr. S. C. Sarin, Chairman

Industrial & Systems Engineering

(ABSTRACT)

This research intends to propose and analyze an integrated algorithm for ATM networks. The integrated algorithm can be divided into three separate policies: buffer management policy, transmission policy, and discard policy. The buffer management policy is presented by previous publications, thus not analyzed in this research. The objective of the transmission policy is to minimize the conflicting goals of minimizing delay and 1-point cdv at an ATM node, while the objective of the discard policy is to distribute loss fairly among connections.

The transmission policy utilizes the time-varying priority concept to assign a priority index to each connection in an ATM node. A priority index of a connection is the weighted sum of the scaled time-varying priorities associated with delay and 1-point cdv for the cell waiting at the head of the connection's queue. The set of priority indices from all connections gives an indication of relative urgency for connections to transmit cells. Thus, the selection of a connection to transmit a cell is simply to choose the connection with the highest urgency as indicated by its priority index.

The discard policy tries to allocate loss of cells fairly among connections. A two-stage approach is proposed. Stage one involves the use of a modified version of generic cell rate algorithm (MGCRA) to enforce incoming cell rates compliance from all connections. Stage two implements a rotation index scheme such that the long term pair-wise cell loss ratios among connections approach the pair-wise ratios of the allocated-CLR's (ACLR).

The behavior of these two policies under Bernoulli distribution are studied and analyzed. A theoretical approximation approach is developed to find the wait time and the cdv-count distributions for a two-connection system example. Simulation is used to studied these two policies under various conditions.

To my parents.

ACKNOWLEDGMENTS

I wish to express my sincere gratitude to my advisory committee, Dr. Sarin, Dr. Kobza, and Dr. Midkiff, for their friendship and inputs to my research. Their guidance and patience provided much inspiration, encouragement, and enlightenment. I would also like to thank Dr. Jacobson for his input on the subject of simulation. I wish to thank Dr. Koelling for granting me the access to the Industrial and Systems Engineering Computer Lab, when I need the computing resources the most. I would like to thank the entire Operations Research faculty for providing an intellectually challenging environment and constant encouragement.

I would like to thank Ms. Lovedia Cole, for her friendship and guidance through all the paperwork. I wish to thank my friends in the Simulation and Optimization Laboratory and the ISE Computer Laboratory for their friendship and encouragement. They made the life as a graduate student fun and memorable.

I wish to thank my parents, my granduncle and grandaunt, and my sisters, for their tireless support and encouragement throughout my graduate education.

TABLE OF CONTENTS

	Page
Abstract	ii
Dedication	iii
Acknowledgments	iv
Table of Contents	v
List of Figures	vii
List of Tables	ix
Chapter 1 Introduction	1
1.1 Asynchronous Transfer Mode (ATM)	1
1.1.1 Characteristics	1
1.1.2 ATM Within B-ISDN	3
1.2 Objective of the Research	4
1.3 Organization of the Thesis	4
Chapter 2 System Overview	5
2.1 Terminology	5
2.2 Reference Model for the User-Network Interface	7
2.3 Reference Model Used for ATM Switches	8
2.4 Functional Description	8
2.4.1 Fair Access Among Connections	9
2.4.2 Requirements Defined in the Traffic Contracts	9
2.4.3 Optional Traffic Shaping	11
2.5 Cell Delay Variations	11
2.6 Generic Cell Rate Algorithm	12
2.7 Model of the Physical Environment	14
2.8 Miscellaneous Topics	15
2.8.1 Allowable Combinations of Traffic Parameters in A Direction	15
2.8.2 "Best Effort" Traffic	16
2.8.3 Sequence of Transmission	16
2.9 Summary	16
Chapter 3 Techniques Proposed In The Traffic And Congestion Control	17
3.1 Rate Throttle Control	17
3.1.1 Leaky Bucket Algorithm	18
3.1.2 Traffic Shaping	19

	Page
3.2 Queue Management Policies	20
3.2.1 Cell Discarding Policies	21
3.2.2 Cell Transmission Policies	23
3.2.3 Integrated Approach	31
3.2.4 Miscellaneous Policies	37
3.3 Summary	38
Chapter 4 Algorithmic Development	39
4.1 Definitions	39
4.2 The Buffer Management Policy	40
4.3 The Discard Policy	41
4.3.1 Discard by MGCRA	42
4.3.2 Discard by Rotation	43
4.4 The Transmission Policy	46
4.5 Approximation of the Wait Time and the cdv-Count Distributions	52
4.6 Summary	68
Chapter 5 Simulation Model And Results	72
5.1 The Simulation Model	72
5.2 Impact of Individual Parameters in Transmission Policy	76
5.3 The Behavior of The Discard Policy	90
5.4 Performance Comparison of Transmission Policies	94
5.5 Performance Comparison of Discard and Buffer Management Policies and Overflow Discard Policy with Fixed-Size Buffers	100
5.6 Summary	101
Chapter 6 Conclusions	102
References	105
Vita	109

LIST OF FIGURES

	Page
Figure 1.1 B-ISDN Protocol Reference Model	3
Figure 2.1 PCR Reference Model	7
Figure 2.2 SCR Reference Model	8
Figure 2.3 Model of A Nonblocking ATM Switch	9
Figure 2.4 Virtual Scheduling Algorithm for GCRA (I, L)	13
Figure 2.5 Continuous-State Leaky Bucket Algorithm for GCRA (I, L)	13
Figure 2.6 Physical Model of Environment	14
Figure 3.1 Model of Buffered Leaky Bucket Algorithm	18
Figure 3.2 Cell Discarding Policies	21
Figure 3.3 Queue Threshold Policy	25
Figure 3.4 Weighted Round-Robin Policy	28
Figure 3.5 Cost-Based Scheduling Policy	34
Figure 3.6 The Head-of-Line Priority Jump Policy	35
Figure 3.7 Static Priority Policy with Loss and Delay Priorities	36
Figure 3.8 Stop-and-Go Queueing Policy	38
Figure 4.1 Locations Where the TTT and the TAT are Defined	47
Figure 4.2 Trade-Off between Delay and 1-Point CDV	49
Figure 4.3 Generic Forms of the Scaled Time-Varying Priorities for Delay and 1-Point CDV	50
Figure 4.4 Possible Scenarios under the Proposed Algorithm	54
Figure 4.5 The Binary Representation of Arrival Pattern	57
Figure 4.6 A Generic Markov State for a Two-Connection System	59
Figure 4.7 Possible States and Priority Indices in a Two-Connection Example	61
Figure 4.8 An Example of Step 5	63
Figure 4.9 State Removal Due To No Transition Probability	64
Figure 4.10 Markov Chain for the Two-Connection System Example	64
Figure 5.1 Simulation Model	72
Figure 5.2 Simulation Results, Run 1A	79
Figure 5.3 Simulation Results, Run 1B	80
Figure 5.4 Simulation Results, Run 1C	81
Figure 5.5 Simulation Results, Run 1D	84
Figure 5.6 Simulation Results, Run 1E, 1F, and 1G, Wait Time Distribution	86

	Page
Figure 5.7 Simulation Results, Run 1E, 1F, and 1G, cdv-count Distribution	88
Figure 5.8 Cell Discard Due To MGCRA	93
Figure 5.9 Cell Discard Due To Rotation	93
Figure 5.10 Simulation Results, Run 3A, Wait Time Distributions	97
Figure 5.11 Simulation Results, Run 3A, cdv-count Distributions	98

LIST OF TABLES

	Page
Table 1.1 ATM Service Classes	4
Table 2.1 Allowable Traffic Rate Combinations	15
Table 3.1 The Calculations of Transmission Sequence for Golden Ratio Policy	29
Table 4.1 Connection Parameters for a Two-Connection System Example	60
Table 4.2 Candidate States	61
Table 4.3 State Solution to The Two-Connection System Example	66
Table 4.4 Comparison of Results for the Two-Connection System Example	67
Table 4.5 Comparison of Results for the Two-Connection System Example	68
Table 4.6 Theoretical Approximation under Different Search Space for Two-Connection System	70
Table 4.7 Theoretical Approximation under Different Search Space for Two-Connection System	71
Table 5.1 Validation of the Simulation Model	75
Table 5.2 Objectives of Simulation Runs	77
Table 5.3 Parameters Used to Study the Effect of Individual Parameter	78
Table 5.4 Simulation Results, Run 1A	78
Table 5.5 Simulation Results, Run 1B	80
Table 5.6 Simulation Results, Run 1C	81-83
Table 5.7 Simulation Results, Run 1D	84
Table 5.8 Simulation Results, Run 1E, 1F, and 1G, Wait Time, Connection 1	87
Table 5.9 Improvement in the Probabilities of Violating the ACDV Specifications	89
Table 5.10 Parameters Used in Simulation Runs for Discard Policy	90
Table 5.11 Simulation Results, Run 2A	91
Table 5.12 Simulation Results, Run 2B	91
Table 5.13 Simulation Results, Run 2C	92
Table 5.14 Simulation Results, Run 2D	93
Table 5.15 Parameters Used to Study Various Policies	94
Table 5.16 Results for the Probabilities that Cells are Compliant with the Traffic Contract	100
Table 5.17 Parameters Used to Compare The Performance of Discard and Buffer Management Policies	100
Table 5.18 Simulation Results, Run 4A and Run 4B	101

CHAPTER 1. INTRODUCTION

Since the early seventies, computer scientists and electrical engineers have been working on the next generation of high-speed network infrastructures and standards capable of handling multimedia information. The goal is to find a new architecture to handle information traffic with various characteristics, such as digital data, video, audio, and digitized images, with minimal loss in the quality of information as it is being transported. Additionally, the new architecture should have the ability to be scaled easily to meet ever-increasing demands [1-7]. The Integrated Services Digital Network (ISDN) protocol model was first published in 1984, which then evolved into other models, including the Broadband Integrated Services Digital Network (B-ISDN) and Frame Relay standards. In the early nineties, the International Consultative Committee for Telephone and Telegraph (CCITT, now known as the International Telecommunications Union, Telecommunications, or ITU-T) and the American National Standards Institute (ANSI) chose Asynchronous Transfer Mode (ATM) as the basis for the future B-ISDN [8]. Currently, the ATM standard is by far the most widely accepted alternative to handle the demanding requirements of multimedia information.

The ATM standard is designed to handle various types of traffic. As a result, the overall system must be able to handle the complex issues of resource sharing and allocation. One of the important issues that remains to be resolved is the intelligent multiplexing and traffic shaping of cells at both the user-network interface and within ATM switches. Current publications tend to present algorithms that optimize a single performance parameter, without regard to the trade-off of other parameters. This research investigates a heuristic scheduling/sequencing algorithm for ATM switches that tries to find a near-optimum solution under competing goals. In Section 1.1, a brief description of the ATM standards is given. The objective of this research is stated in Section 1.2. The organization of the thesis is then explicated in Section 1.3.

1.1. Asynchronous Transfer Mode (ATM)

1.1.1. Characteristics

Unlike previous communication architectures, which were designed with particular applications in mind, ATM is designed to handle multiple applications, including possible future applications. Asynchronous transfer mode is an information transfer mode based on the concepts of cell-based switching, statistical multiplexing, connection-oriented services, and service performance guarantees. Each of these characteristics are discussed further below.

Cell-Based Switching

The basic information unit in ATM networks is a 53-byte *cell*. Each cell consists of two parts: header and payload. The header is five bytes long and contains information related to routing, flow control, payload, loss priority, and error checking. The benefit of cell-switching versus packet-switching is that all of the information units are of the same length, thus all information units receive an equal amount of system resources, i.e., buffer space and server processing time. The drawbacks are that the efficiency may suffer as the payload has to be stuffed with fillers if the information to be sent requires less than 48 bytes, and high overhead due to small cell size.

Statistical Multiplexing

Different types of information sources generate different traffic demands, and many traffic patterns are bursty in nature. Asynchronous transfer mode utilizes the on-off nature of bursty traffic patterns to allow for more information sources to share a common medium.

Connection-Oriented Services

In traditional connectionless packet switching, the route that each packet travels through the data network may not be the same. This phenomenon creates variations in the propagation delays and packets may arrive out of sequence, thus additional processing may be required to reassemble data at the destination.

Asynchronous transfer mode, on the other hand, provides a connection-oriented service. Prior to any transmission of information, a connection setup is required. A good example of a connection-oriented service is the telephone network. During a connection setup, the route that information travels through a network is determined. If the current system cannot support the resource requirement that a user requests, either the user modifies his or her request or a connection is denied.

Service Performance Guarantee

During connection setup, the Quality of Service (QoS) parameters, the traffic descriptors, and the class of service are defined. The Quality of Service parameters are used to define the maximum and minimum performance commitments the ATM network guarantees. Some examples of QoS parameters are delay, cell loss ratio, and cell delay variation. The different classes of services are described in Section 1.1.2. The

traffic descriptors used are determined based on the class of service specified. Examples of the traffic descriptors are peak cell rate and sustainable cell rate.

1.1.2. ATM Within B-ISDN

The reference model of B-ISDN adopted by ITU-T is illustrated in Figure 1.1 [9].

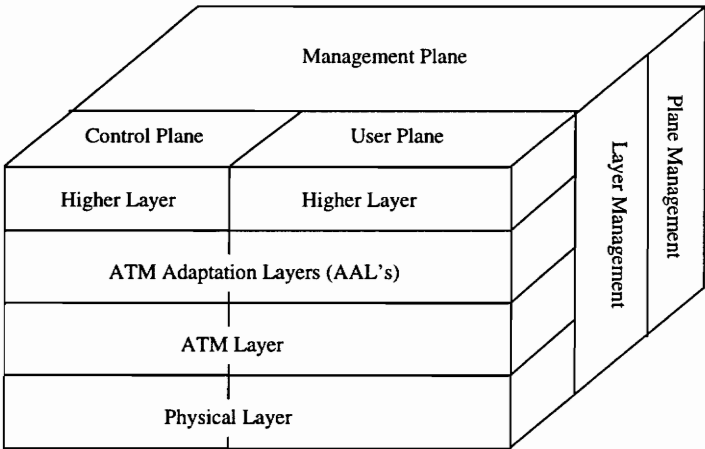


Figure 1.1 B-ISDN Protocol Reference Model

As depicted in the figure, ATM includes the ATM adaptation layers (AAL's) and the ATM layer in the B-ISDN reference model. The ATM layer corresponds to the data link layer and a portion of the physical layer in the Open System Interconnect Reference Model (OSIRM), while the AAL's correspond to a portion of the network layer. The function of the ATM layer is to handle tasks related to construction, extraction, multiplexing, and routing, of cells. The ATM layer also supports usage parameter control (UPC), call connection functions, flow control, and congestion control. The AAL's are implemented to handle different classes of services. Currently, there are four service classes and five AAL's. Table 1.1 [10] depicts the relationship between the five AAL's and the four quality of service (QoS) classes with their respective attributes.

Table 1.1 ATM Service Classes

	Service Classes			
Attribute	Class A	Class B	Class C	Class D
Timing Between Source and Destination	Required		Not Required	
Bit Rate	Constant	Variable		
Connection Mode	Connection-Oriented		Connectionless	
AAL's	AAL1	AAL2	AAL3/4 or AAL5	AAL3/4 or AAL5
Example Application	DS1, E1, nx64 kbps emulation	Packet Video, Audio	Frame Relay, X.25	IP, SMDS

1.2. Objective of the Research

In past publications, the scheduling algorithms or queuing policies proposed tend to emphasize the optimization of a single parameter, i.e., delay or loss ratio. The objective of this research is to propose and analyze an integrated algorithm to be used in ATM switches and ATM adapter cards. The algorithm is designed to seek a good compromise among conflicting goals of minimizing delay, minimizing cell delay variations generated by a communication node, and being fair to information sources in the allocation of system resources and losses. The performance of the proposed algorithm will be tested via simulation, and is compared with the first-come-first-serve (FCFS) and the weighted-round-robin (WRR) algorithms.

To accomplish the desired objective, the research will include a survey of techniques used currently, the physical system description, the algorithm formulation, and simulation analyses of the behavior and comparative performance of the algorithm.

1.3. Organization of the Thesis

The thesis is divided into the following chapters: Chapter 2 contains a brief overview of the ATM standards related to the research; Chapter 3 presents a survey of the current techniques used in congestion and flow control; Chapter 4 explicates the mathematical formulation of the algorithm; Chapter 5 examines the simulation results; and Chapter 6 summarizes the results and suggests future research directions.

CHAPTER 2. SYSTEM OVERVIEW

This research intends to study the behavior of the proposed algorithm to be used in ATM adapters and switches. Before the actual algorithm is proposed, it is necessary to understand the system under investigation. In this chapter, a detailed description of variables in the ATM standard that are related to this research is developed. This chapter is divided into nine sections: Section 1 presents the definition of the terminology used, Section 2 illustrates the reference model for the user-network interface as specified by the ATM Forum, Section 3 illustrates the reference model for the ATM switches, Section 4 examines the requirements on the new algorithm, Section 5 provides the official definition of cell delay variation (CDV), Section 6 depicts the fundamentals of the generic cell rate algorithm (GCRA), Section 7 presents a physical model of the system, Section 8 covers several miscellaneous topics, and Section 9 provides a brief summary of Chapter 2.

2.1. Terminology

The following terms are commonly used to describe ATM network:

Connections: ATM networks are connection-switched networks. Before any information can be transmitted, connections need to be set up between origins and destinations. A good example of a connection-switched network is the telephone network.

Traffic Contracts: During connection setup, a traffic contract is negotiated between the ATM network and the user. The contract contains information such as quality of service parameters, the classes of connections, etc. These parameters essentially are promises made by the ATM network regarding the level of service that is guaranteed. It is also used by the network as a basis to determine whether users are utilizing more resources than claimed.

Traffic Shaper: These are modules used to transform a given traffic pattern into a different one. The main purpose of a traffic shaper is to make certain that the outgoing information from the user's end is in compliance with the traffic contract.

MUX: This is a short form for *multilplexer*. This is essentially a logical selector that has multiple inputs and a single output. An input is selected for output based on a pre-determined logic.

Theoretical Arrival Time (TAT): Theoretical arrival time is the time, on average, at which a cell should arrive at a receiving node in a connection. This is a value calculated based on the average interarrival time, and is used mainly as a reference when checking a connection's transmission rate.

Generic Cell Rate Algorithm (GCRA): This is a generic algorithm used to check conformance of cells and/or to shape traffic. There are two equivalent forms of GCRA, the continuous-state leaky bucket algorithm and the virtual scheduling algorithm. The GCRA is usually written as GCRA(I, L), where I is the increment and L is the limit. The increment is defined as the average amount of time between transmissions and is used in the calculation of TAT; the limit is defined as the maximum variation allowed from the TAT. The two equivalent forms of GCRA are presented in Section 2.6.

Usage Parameter Control (UPC): This is the portion of an ATM network which polices the incoming traffic to make sure that it is compliant with the agreed traffic contract.

Cell: A cell is the smallest unit of information in ATM networks. It consists of 53 bytes of data.

Virtual Channel (VC): A virtual channel is the smallest unit of a connection.

Virtual Path (VP): Multiple virtual channels can be grouped together to form a virtual path. The purpose of virtual paths is to simplify the task of routing multiple connections, if these connections have the same destination.

Peak Cell Rate (PCR): The highest rate of transmission allowed by a connection. It is usually defined as the inverse of the *peak emission interval* (T).

Sustainable Cell Rate (SCR): This is an optional parameter specified by users in their traffic contracts. If the traffic is very bursty, it is not efficient to allocate bandwidth based on PCR. In that case, SCR is used in conjunction with PCR. Sustainable cell rate is defined as the average transmission rate over the life of the connection required by users. Like PCR, the average interval between transmission for SCR is defined as the inverse of SCR, and denoted as T_s .

Cell Delay Variation (CDV): This is the variation of spacing in time between cell arrivals. The CDV used for PCR is denoted as t , while that for SCR is denoted as t_s . More detailed treatment of the CDV is presented in Section 2.5.

Cell Priority: Within each connection, cells can be designated with a cell priority via the use of the *cell priority bit* (CPB). The possible values are 0 and 1, where 1 indicates a lower priority. ATM networks try to discard cells with CPB value of 1 before cells with CPB of 0.

Tagging: Users can control the cells to be discarded by setting a cell's CPB to 1. This action of setting the CPB value is called "tagging."

2.2. Reference Model for the User-Network Interface

In the ATM Forum's User-Network Interface 3.0 [11], the peak cell rate reference model used for terminal equipment is illustrated in Figure 2.1.

At the user-network interface, an integrated algorithm should include functions of the MUX and the traffic shaper. Since the shaper is an optional requirement, it should be allowed to be turned off if the users do not specify its usage.

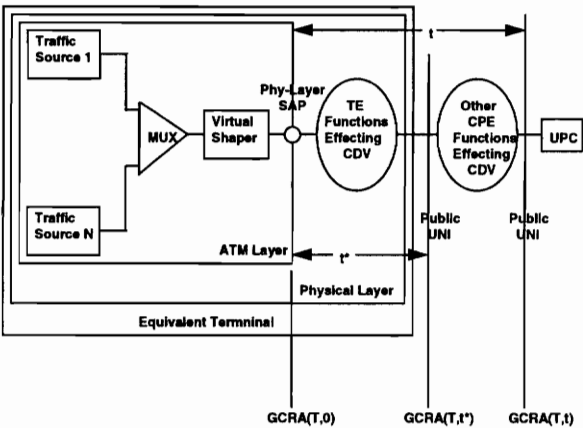


Figure 2.1 PCR Reference Model

The reference model based on the sustainable cell rate is similar to the PCR reference model, and is illustrated in Figure 2.2. It differs from the PCR reference model only in the values used by the GCRA's. Since the roles of the MUX and traffic shaper are exactly the same for both reference models, this research will only consider the PCR reference model.

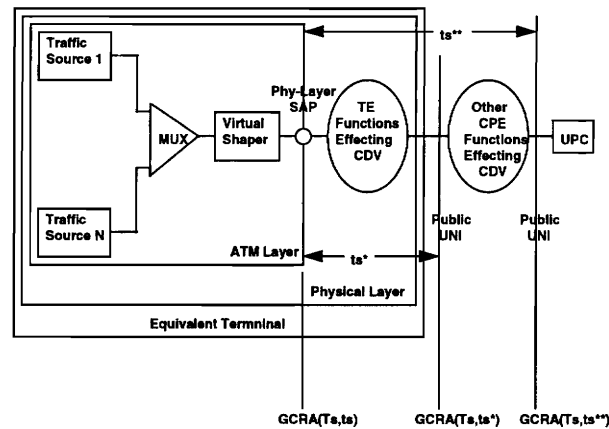


Figure 2.2 SCR Reference Model

2.3. Reference Model Used for ATM Switches

For ATM switches, there are multiple ways of buffering information as it traverses through the switches. The generic reference model is illustrated in Figure 2.3 [12]. Depending on the locations of buffers, the buffering methods can generally be classified into four classes: input buffering, output buffering, internal buffering, and shared buffering [11]. According to past publications, the consensus is that the output buffering and shared buffering provide the optimal implementation among the four alternatives [13-15]. However, a simulation study of input/output buffering under bursty traffics presented in [16] disagrees with this consensus. In this study, an ATM switch with output buffering is assumed.

2.4. Functional Description

An integrated algorithm must perform the following required functions:

1. Ascertain that fair access to system resources is provided to each information source, and

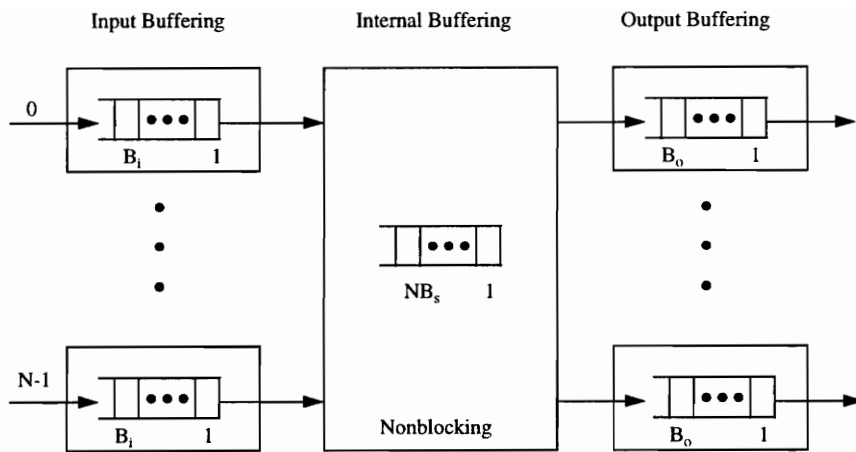


Figure 2.3 Model of A Nonblocking ATM Switch

2. Ascertain that the minimal and maximal requirements defined in the negotiated traffic contracts are satisfied.

An integrated algorithm optionally needs to perform traffic shaping to minimize the probability of a cell being discarded by the network due to non-compliance with traffic contract.

2.4.1. Fair Access Among Connections

It is conceivable that some information sources, maliciously or unintentionally, may try to take advantage of the integrated algorithm implemented in ATM switches, so as to gain a larger share of a system's resources at the expense of other rule-abiding information sources. It is important for an integrated algorithm to have capabilities to handle misbehaved sources.

This topic will be covered more thoroughly in Chapter 3.

2.4.2. Requirements Defined in the Traffic Contracts

Parameters are defined in a traffic contract, which may be used at the interface between a local user and an ATM network, and within the network. These parameters are used as minimum and maximum performance

metrics that ATM networks promise to users. Also, traffic contracts provide a basis on which ATM networks can regulate the information flow and allocate system resources in order to satisfy the commitments made to all connections.

Currently, the performance parameters used in traffic contracts are still being developed. The parameters listed below have been defined by the ATM Forum. These parameters are listed according to their respective purpose.

<u>Name</u>	<u>Used To Measure</u>
Cell Error Ratio	Accuracy
Severely-Errored Cell Block Ratio	Accuracy
Cell Mis-insertion Rate	Accuracy
Cell Loss Ratio (CLR)	Dependability
Cell Transfer Delay (CTD)	Speed
Mean Cell Transfer Delay (MCTD)	Speed
Cell Delay Variation (CDV)	Speed

The parameters of interest in this research are the ones related to dependability and speed. Accuracy is not considered as it is a function of hardware dependability and implementation. The following subsections provide the mathematical definitions for CLR, CTD, and MCTD. The definition of CDV is discussed in more details in Section 2.5.

Cell Loss Ratio

For an ATM connection, the cell loss ratio is defined as the ratio of the number of cells lost and the total number of cells transmitted, or

$$\text{CLR} = \frac{\text{Lost Cells}}{\text{Total Transmitted Cells}}$$

Cell Transfer Delay

The cell transfer delay is defined as the time a cell takes to travel between two measurement points, e.g. the source and destination user-network interfaces. The CTD consists of the propagation delay and the

processing delay. Since the propagation delay is fixed once a connection is determined, the only parameter of interest is the processing delay. Since the CTD is defined as an end-to-end parameter, the value itself is not meaningful at the node level of a network; however, the magnitude of CTD, together with the number of nodes in a connection, can be used to indicate the strictness of the delay requirement in a connection.

Mean Cell Transfer Delay

The mean cell transfer delay is defined as the arithmetic average of a specified number of CTD's of one or more ATM connections. Since the MCTD is an arithmetic average of CTD's, minimizing individual CTD's will minimize the MCTD. Thus, in this study, MCTD will not be used.

2.4.3. Optional Traffic Shaping

Traffic shaping is optional in ATM networks. In an ATM network, all incoming traffic is checked for compliance with the traffic contract. If a particular connection violates its contract, the network has the option of discarding cells it decides to be non-compliant. If users choose not to shape their traffic and transmit cells to the ATM network, then the network determines arbitrarily the cells to be dropped. Thus, to ensure the quality of transmission, users may want to shape their data before transmitting, so they can ascertain that only cells with lesser importance are likely be discarded. The algorithm used for traffic shaping is not standardized and is left to individual vendors.

2.5. Cell Delay Variations

According to the ATM Forum [17], there are two definitions for cell delay variation: 1-point CDV and 2-point CDV. The calculation of 1-point CDV requires only information available at a network node, while the 2-point CDV requires measurements from two different nodes. Since this research is interested in developing an algorithm to be used within a communication node, the 1-point CDV is used and its definition is presented.

The 1-point CDV is defined as the difference between the reference arrival time and the actual arrival time for each cell at a node. Mathematically, let y_k be the CDV of cell k , c_k be the reference arrival time of cell k , and a_k be the actual arrival time of cell k , then

$$y_k = c_k - a_k$$

The reference arrival time of cell k is calculated as follows:

$$c_0 = a_0 = 0$$

$$c_{k+1} = \begin{cases} c_k + T & \text{if } c_k \geq a_k \\ a_k + T & \text{otherwise} \end{cases}$$

where

T is the peak emission interval.

A positive CDV (y_k) value indicates cell clumping, while a negative value indicates cell dispersion. Normally, ATM networks are not concerned with negative CDV, as it indicates a connection is transmitting at a slower rate than the one specified in the traffic contract. On the other hand, if cells from a connection persistently have positive CDV's, the connection may violate the contract and appropriate actions may need to be taken.

2.6. Generic Cell Rate Algorithm

The generic cell rate algorithm (GCRA), as defined in ATM UNI 3.0, can be used to check the conformance of a connection's traffic with respect to its traffic contract. Thus, GCRA may be used as a form of usage parameter control (UPC). However, an ATM network vendor is not required to use GCRA as defined in [18], and may use any UPC as long as its operation "does not violate the QoS objectives of a compliant connection" [18].

A GCRA is used to define the relationship between the peak cell rate and the cell delay variation, and between the sustainable cell rate and the burst tolerance. Two parameters are used in a GCRA: increment (I) and limit (L). The notation of $GCRA(I, L)$ is used to indicate a generic cell rate algorithm with an increment value of I and a limit value of L . For example, at the public UNI in the PCR reference model in Figure 2.1, the increment is the peak emission interval (T), and limit is the CDV (t).

There are two equivalent forms of GCRA: virtual scheduling algorithm and continuous-state leaky bucket algorithm. The algorithms are illustrated in Figure 2.4 and Figure 2.5, respectively.

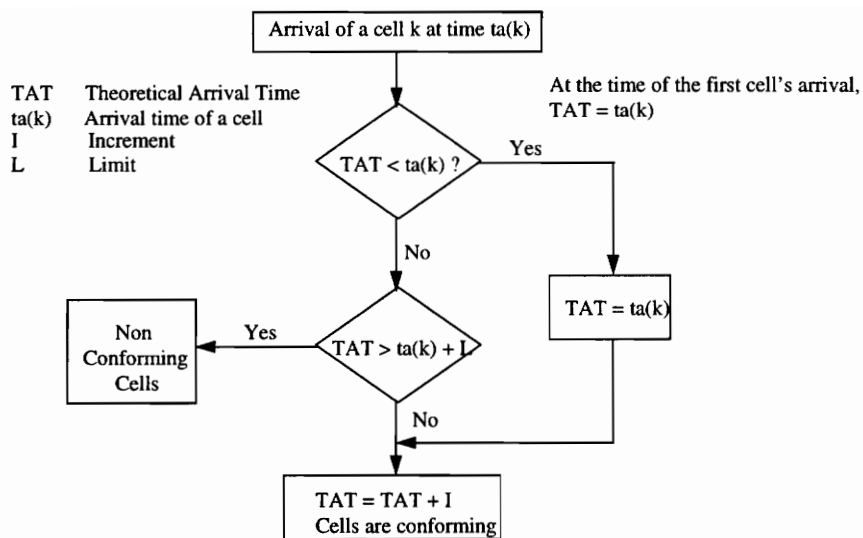


Figure 2.4 Virtual Scheduling Algorithm for GCRA(I, L)

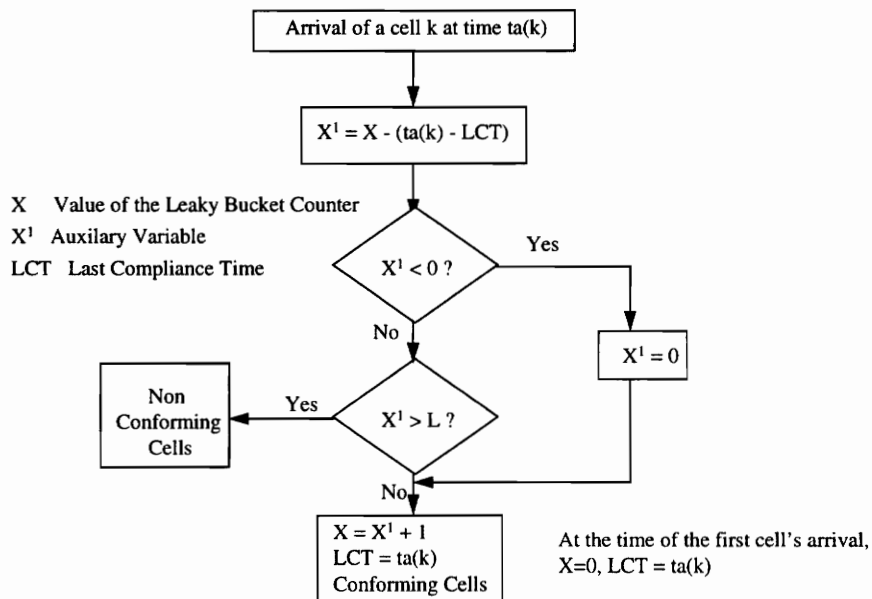


Figure 2.5 Continuous-State Leaky Bucket Algorithm for GCRA(I, L)

2.7. Model of the Physical Environment

A representation of the physical environment in an ATM system in which the integrated algorithm operates is depicted in Figure 2.6.

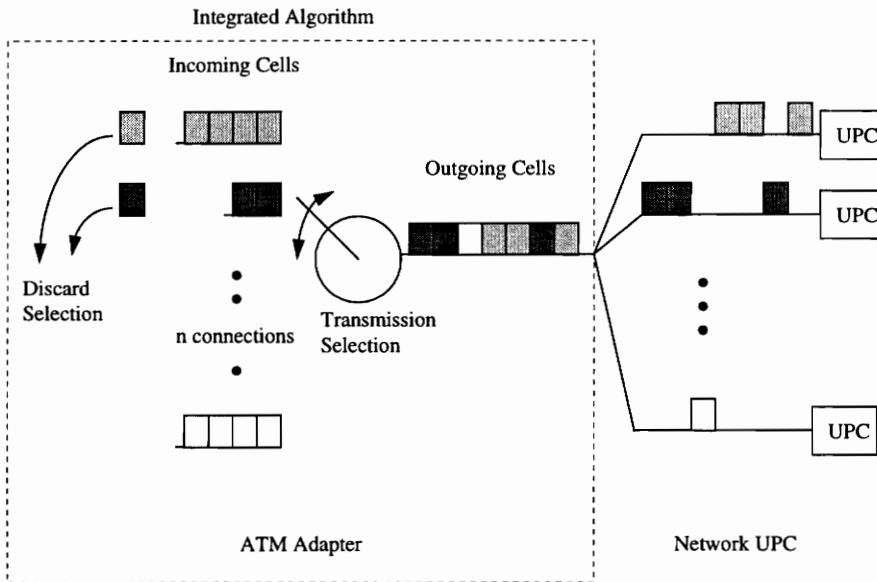


Figure 2.6 Physical Model of Environment

The model utilizes the *slotted time* concept, where a time line is divided into discrete slots. Each slot represents a *cell time*, or the amount of time required by the transmitter to send a cell. A cell time slot is the smallest unit of time in the model. For example, if the outgoing link has a bandwidth of 155 Mbs, then a cell time is about 2.7 micro-seconds ($= (53 \text{ bytes/cell} \times 8 \text{ bits/byte}) / (155,000,000 \text{ bits/second})$).

There are n connections in the system. The number of connections is a function of time, and varies as connections are setup and disconnected at different time. Each connection has a buffer which accepts cells as they arrive. If a buffer is full when cells arrive, then the cells are dropped according to the discard policy. During each cell time slot, only a cell can arrive in each connection. The burst length, or number of cells which arrive back-to-back, depends on the connection's characteristics. Only one cell is transmitted during each time slot from the transmitter. Thus, the sequence in which the cells leave the transmitter consists of a mixture of cells from the n different connections. The cell to be transmitted next is selected based on the transmission policy. Traffic-shaping functions for each connection should be implemented within the transmission policy. The network UPC polices the traffic of each connection; thus conceptually, the outgoing stream of cells is broken into individual connections, upon which the UPC applies the GCRA.

2.8. Miscellaneous Topics

In this section, three miscellaneous topics related to the system under investigation are described: the allowable combinations of traffic parameters, the best effort traffics, and the sequence of transmission.

2.8.1. Allowable Combinations of Traffic Parameters in A Direction

Table 2.1 shows the valid combinations of traffic parameters. Note that for each connection, a maximum of two transmission rates can be specified for each connection. The traffic in a connection is divided into two parts based on the cell priority bit. The notation CLP=0 denotes that the portion of traffic which has its CPB's set to 0; CLP=0+1 denotes the sum of traffic with CPB's of 0 and 1, or all traffic. Thus, each transmission rate defined for a connection covers the portion of traffic specified by the CLP. For example, if a row in Table 2.1 has "Peak Cell Rate, CLP=0", this means that the peak cell rate is defined over the cells

Table 2.1 Allowable Traffic Rate Combinations

#	Allowable Combination of Traffic Parameters	
1	Peak Cell Rate	CLP=0
	Peak Cell Rate	CLP=0+1
2	Peak Cell Rate	CLP=0
	Peak Cell Rate	CLP=0+1
	Tagging = tagging requested	
3	Peak Cell Rate	CLP=0+1
	Sustainable Cell Rate	CLP=0
	Maximum Burst Size	CLP=0
4	Peak Cell Rate	CLP=0+1
	Sustainable Cell Rate	CLP=0
	Maximum Burst Size	CLP=0
	Tagging=tagging requested	
5	Peak Cell Rate	CLP=0+1
6	Peak Cell Rate	CLP=0+1
	Sustainable Cell Rate	CLP=0+1
	Maximum Burst Size	CLP=0+1

in the traffic with their CPB's set to 0.

To simplify the analysis, the algorithm is designed for combination 5. However, with slight modifications, the proposed algorithm can be applied to the other combinations as well.

2.8.2. “Best Effort” Traffic

The ATM standards allow “best effort” connections. The implication of a “best effort” connection is that the ATM network does not guarantee the performance of the connection. Connections with “best effort” designation do not necessarily have constant traffic parameters associated with them. The only traffic parameter declared in the contract for a best effort connection is PCR for CLP=0+1. However, as the name implies, the performance that users get from a best effort connection depends on the availability of network resources at any given time.

This research will not consider best effort traffic.

2.8.3. Sequence of Transmission

The sequence of cells within a connection has to be maintained from beginning to end. Thus, the sequence of transmission for a connection from a node has to be the same as the sequence of arrivals to the node. From Figures 2.1 and 2.2, each connection may have a queue associated with it. Within each queue, the first-come-first-serve (FCFS) policy has to be maintained. Thus, for each connection, a first-in-first-out (FIFO) policy is strictly observed.

2.9. Summary

In this chapter, a detailed description of the system under investigation is developed. Various pertinent terms are defined. The reference models used for the user-network interface and within the ATM switches are described. The requirements of the proposed algorithm are also examined. The definitions of cell delay variations and the generic cell rate algorithm are outlined, and some miscellaneous topics related to the system under study are also described.

CHAPTER 3. TECHNIQUES PROPOSED IN THE TRAFFIC AND CONGESTION CONTROL

Due to the high transmission rate of ATM networks, the impact of processing delay has risen significantly, and the traditional reactive, closed-loop traffic and congestion control schemes no longer work well. As links in a network become capable of transmitting increasingly larger amount of information in a time period, the processing time allowed at each node so congestion does not occur decreases. The traditional closed-loop control is designed based on the assumption that a system's status does not change much during the round-trip propagation delay of controlling messages. However, the high transmission rate decreases the amount of reaction time a reactive control scheme has. Thus, it becomes very likely the closed-loop control is exerted after the system's status has changed. In response to the deficiency of these traditional techniques, open-loop control and input rate regulation schemes have been proposed and analyzed in the past few years. The approaches used can be roughly divided into two groups: (i) rate throttle control, and (ii) queue management. The rate throttle control approach emphasizes the preventive congestion control via regulation of the transmission rate at the user-to-network interface or the network-to-network interface, while the queue management approach tries to optimize the performance of local nodes by minimizing the effects of processing delay and local congestion.

In this chapter, the literature related to the proposed research topic is reviewed. The chapter is organized into three sections: Section 3.1. discusses the evolution of rate throttle control schemes; Section 3.2. examines queue management policies proposed in the past; and Section 3.3. gives a summary of the chapter. Due to a vast amount of work in this area, the review is by no means exhaustive, but it includes pertinent papers that represent available knowledge on the proposed areas of research.

3.1. Rate Throttle Control

In 1988, Woodruff, Rogers, and Richards [19] presented a framework of congestion control in high-speed packetized networks. The main idea is to use *access control* to prevent congestion from occurring, rather than to react to congestion after it has already taken place. Also suggested in [19] was that traffic shaping should be used to monitor and enforce bandwidth commitment. In 1989, Eckberg, Luan, Lucantoni [20] proposed a *Bandwidth Management* scheme for congestion control, in which a leaky bucket algorithm (LBA) was suggested as a good way for bandwidth enforcement. The rest of this section is divided into two

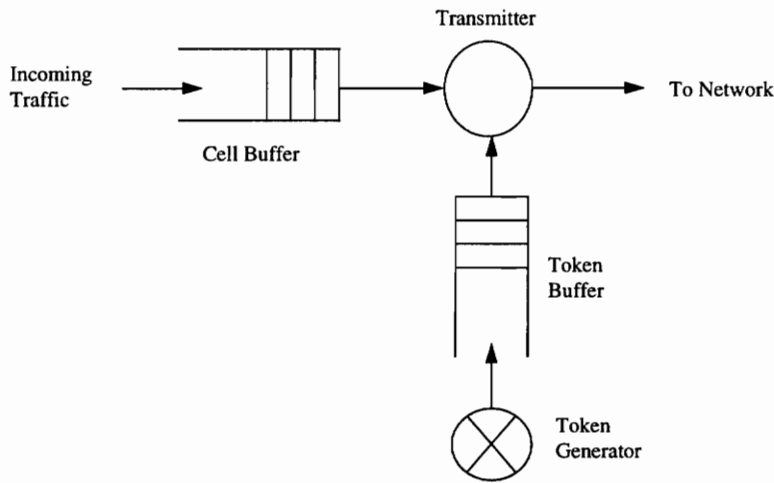


Figure 3.1 Model of Buffered Leaky Bucket Algorithm

parts: a review of the LBA is presented in the section 3.1.1., while a review of the traffic shaping is given in section 3.1.2.

3.1.1. Leaky Bucket Algorithm

The model of a buffered-LBA is illustrated in Figure 3.1. The basic operation of a LBA is now described. Cells arrive at the cell buffer which has a first-come-first-served policy. The interarrival time distribution of arriving cells depend on the types of traffic. Cells which arrive after the cell buffer is full are discarded. Thus the size of the cell buffer affects the loss probability of the traffic. The transmitter can serve only one cell at a time. For a cell to be transmitted, there must be at least one token available in the token buffer. Tokens are generated at a pre-defined rate; normally, either the peak cell rate or the sustainable cell rate. If the token buffer is full, the token generator is shut-off. The size of the token buffer determines the degree of cell clumping and the dispersion of the outgoing traffic. Under the LBA, if cells arrive at a rate lower than the token generation rate, a token is always available, and cells can be transmitted; if cells arrive too fast, after exhausting the tokens stored in the token buffer, the maximum transmission rate is limited by the token generation rate.

In 1990, Sidi, Liu, Cidon, and Gopal [21] credited Turner [5] for proposing the original LBA, and presented an analysis of the buffered-LBA under a Poisson arrival assumption. Berger [22] suggested that

the throughput and blocking of incoming jobs were functions of both the buffer size and token size. Bala, Cidon, and Sohraby [23] proposed and analyzed a tagging scheme to allow a transmission rate above the token generation rate when networks were under-utilized. Ahamdi, Guérin, and Sohraby [24] analyzed the LBA under the assumption of a batch arrival process. In 1994, Sohraby and Sidi [25] analyzed the LBA under the assumption of a general discrete Markovian arrival process. Wu, Lambadaris, Lee, and Kaye [26] presented a simulation study of the LBA with various combinations of adaptive, selective cell marking and priority policies for a mix traffic of data and voice. The distribution used for data source was a Poisson batch arrival with a geometrically distributed batch length, and the distribution used for voice traffic was a Poisson arrival process with an exponentially distributed length .

In 1994, Kuang [27] provided a general proof of the variance reduction property of the LBA. In another independent study, Liu and Towsley [28] presented a mathematical analysis of the burst reduction properties of rate-control throttles. Also in the same year, Anantharam and Konstantopoulos [29] showed that given the token buffer size C , the burstiness of the output flow is a function of C and that burstiness increases with C .

3.1.2. Traffic Shaping

There are two main reasons for the use of traffic shapers: to be compliant with the Traffic Contracts and to increase statistical gains of ATM connections. For bursty traffic, it is likely that the actual peak cell rates may be higher than the ones specified in the traffic contract due to inaccurate estimation of traffic characteristics. In this case, traffic shaping is used to smooth out the peak demand, so that the connections stay compliant with the traffic contract. As a matter of fact, the LBA has been suggested to be a good shaper due to its smoothing effect [30]. Also, it has been shown that for a given set of QoS values, the statistical multiplexing gain increases as the peak bit rate decreases, even though the average bit rate and the total traffic offered remain the same [31].

The implementation of a traffic shaper is usually a function of the queuing discipline used. According to [30], there are the following eight common methods to shape traffic: buffering, spacing, peak cell rate reduction, scheduling, burst length limiting, source rate limiting, priority queuing, and framing. Among the eight alternatives, buffering, spacing, scheduling, and priority queuing can be considered as queue management policies. Peak cell rate reduction and burst length limiting require information sources to transmit at a rate or a maximum burst length smaller than the ones specified in the traffic contract; however, these two methods raise the question as to why smaller parameters are not selected when the traffic

contracts are negotiated. Source rate limiting exploits the inherent nature of connections, such as DS1 emulation, so that the transmission rate is bounded by default; this method only works if there are special features within the system, such as standardized transmission rates, that can be exploited. Framing is the only other viable alternative which allows active management of information flows. Since the traffic shaping functions are essentially the by-products of the queuing policies or flow control techniques, the details of various algorithms are treated fully in Section 3.2.

3.2. Queue Management Policies

Although the buffered-LBA has been established as the *defacto* throttle regulation scheme, there have been some publications depicting scenarios where the LBA has failed [32, 33]. There are two problems associated with the source rate throttle control scheme:

1. The analysis of the rate throttle control scheme does not account for the aggregate behavior of traffic from multiple sources. As pointed out in [32], even though all traffic transmission rates are regulated, it is still possible to have congestion within the network due to simultaneous peak demands.
2. The buffer requirement may be exceedingly large for connections that have high transmission rate and are loss sensitive. Reference [33] presents scenarios in which the LBA was not adequate to satisfy the QoS commitment without large buffering at multiplexing and demultiplexing points of networks.

The examples used in [32, 33] illustrate the need for a different form of queue management than the traditional *first-come-first-served* (FCFS) policy, and integration of MUX and queue operations.

Due to the diverse traffic characteristics in ATM networks, multiple connections sharing the same resources may not have the same requirements on delays, loss ratios, and cell delay variations. Various policies designed to function in a heterogeneous environment have been proposed and analyzed. This section is divided into four subsections based on the functionality of the policies: Section 3.2.1 describes the cell discarding policies proposed in the past; Section 3.2.2 examines the cell transmission policies; Section 3.2.3 presents the integrated approaches proposed; and Section 3.2.4 covers the miscellaneous policies.

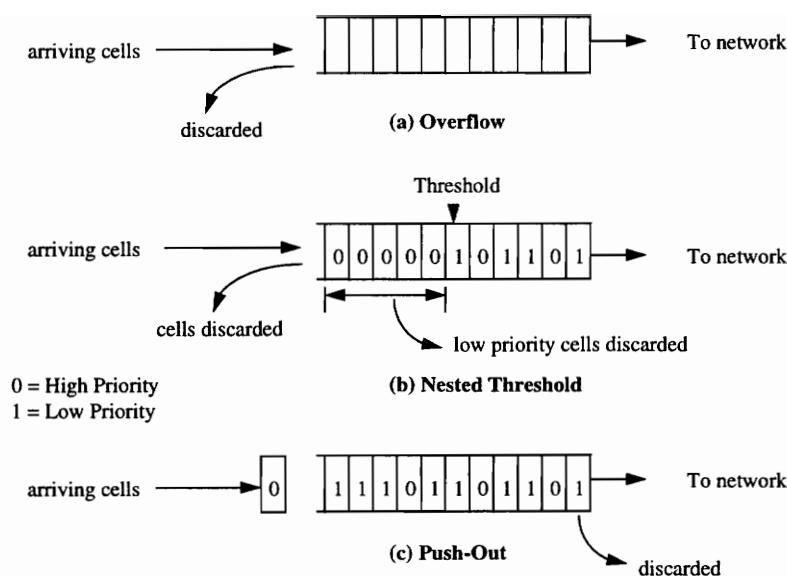


Figure 3.2 Cell Discarding Policies

3.2.1. Cell Discarding Policies

Cell discarding policies determine how cells are dropped if a local queue experiences congestion. Generally, there are three common techniques: overflow, nested threshold, and push-out. These policies are depicted in Figure 3.2, and described in the following paragraphs. In some cases, when tagging is allowed and multiple priority buffers are available, cell discard policies may be modified so that when a buffer becomes full, a cell's priority is changed and sent to a different buffer which is not full [23]. To simplify the presentation, the tagging scenario is considered as a variant of the three cell discarding policies, and not discussed.

The overflow policy is the simplest discarding policy: any cell arriving after a buffer is full is discarded, regardless of its priority. The overflow policy can be considered as the traditional cell discard method. The advantage of this policy is its simplicity; however, connections requiring low loss may suffer loss in greater proportions than the proportions indicated by the specified CLR's. The overflow policy is depicted in Figure 3.1 (a).

A nested threshold (NT) [34, 35] policy decides which cells to drop based on the available space in buffer and the cells' priorities. Under this policy, the buffer space is partially shared. Figure 3.2 (b) depicts how

the buffer is partially shared. In an ATM cell, a one-bit field, known as the *cell priority bit* (CPB), is used to indicate the priority of the cell. Thus, a stream of cells can be divided into a high priority stream and a low priority stream based on the settings of CPB's. The buffer is divided into two parts by a threshold marker. Before the threshold is reached, high and low priority cells share the buffer equally; once the threshold is reached, only high priority cells are allowed to enter the buffer. When the buffer is full, all arriving cells are dropped.

Theoretically, the NT policy is not limited to only two classes of traffic. Bommel and Ilyas [36] have proposed a modification to the ATM cell structure so that four priorities are possible. Under the proposed cell structure, there will be three threshold markers to divide the buffer into four sections, where the first section is shared by all four classes of traffic, the second section allows only cells with the three higher priorities to enter, the third section permits only the cells with two highest priorities, and the last section allows only the highest priority cells. According to [36], NT provides the best performance for queuing of traffic with multiple priorities. However, NT has an implementation problem of determining the locations of optimal thresholds in a buffer, when ATM connections have diversified requirements.

The push-out policy [36, 37] discards a cell based on its priority. The buffer is completely shared for all cells. If a high priority cell arrives and finds the buffer full, a cell with low priority is dropped. The low priority cell to be discarded may be selected randomly, based on a first-come-first-drop policy, or a number of other possible schemes. If a cell arrives when the buffer is filled with only high priority cells, the arriving cell is dropped.

The PO policy is used to protect high priority traffic from congestion, at the expense of low priority traffic. Figure 3.1 (c) illustrates PO with the first-come-first-drop for the low priority cells. It is pointed out in [36] that the PO policy provides the best protection for the high priority traffic. The PO policy also allows sharing of the buffer by all cells, therefore buffer space is utilized fully [38].

The PO and the NT policies differ in the management of low priority cells. In a PO policy, low priority cells are discarded regardless of where they are in the buffer. Thus the high priority cells can bump the low priority ones out of the queue as they arrive. It is entirely possible that every low priority cell is discarded to make room for the high priority cells. Hence, the performance of the low priority cells depends on the behavior of the high priority cells: during the period when there are a lot more of the high priority cells than the low priority cells, the low priority cells suffer greatly; on the other hand, if the number of high priority cells is small, the low priority cells have a much better performance. The NT policy allows the low priority

cells to stay in the buffer once they are admitted and manages priorities via threshold markers. The performance of the low priority cells depends on the amount of space allocated for each threshold and the behavior of the high priority cells.

3.2.2. Cell Transmission Policies

The cell transmission policies determine which connection is allowed to transmit in a multi-connection environment. The transmission policies affect the delay and CDV, and determine how each connection's outgoing traffic behaves. In the past, many policies have been proposed and analyzed; three common schemes are reviewed in the following paragraphs: first-come-first-served, priority, and interleaving. In the following pages, these three schemes and some variations are discussed.

First-Come-First Served Policy (FCFS)

The first-come-first-served policy is usually used as the baseline policy in an analysis. Under this policy, cells are transmitted based on their arrival time. Some additional policies may be required to resolve conflicts if multiple cells arrive at the same time. The FCFS policy is easy to implement and analyze, but it does not differentiate the diverse requirements of different classes of traffic since only arrival times are used to determine the transmission sequence.

Priority

The priority policy transmits cells with higher priorities first. The definition of priority depends on the goals to be optimized. For example, the priority used in a dynamic priority scheme (to be discussed) can be defined in terms of the remaining time before a deadline or the remaining buffer space. Generally, the priority policies can be divided into two classes: static priority and dynamic priority.

Under the static priority scheme [39-41], the priority of cells does not change with time. Any cell with a high priority will be transmitted first. A major drawback of the static priority policy, as pointed out in [42], is that it does not consider the varying urgency of cells with respect to time. A good example of the static priority schemes is to give a real-time application, such as teleconference connections, a high priority, while data applications, such as file transfer, a low priority.

Priorities should be assigned with consideration on the nature of connections. In the example given, the teleconference application is given the high priority, while file transfer is given the low priority. This may pose problems as teleconference applications usually can tolerate a high cell loss rate, while file transfer does not permit any loss of data. If loss of cells occurs for data applications because buffers are full, these applications must request a re-transmission of the lost data, which can further aggravate the congestion in a local node. To solve the dilemma, Chao and Uzun [43] proposed a priority matrix scheme that ranked each connection based on its delay priority and loss priority. The scheme is presented in Section 3.2.3.

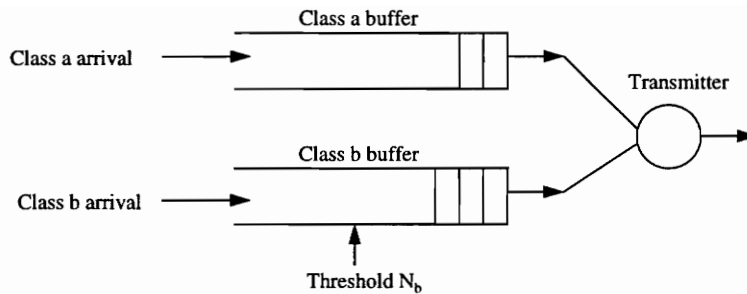
In view of the rigid nature of the static priority policies, dynamic priority schemes are proposed. Dynamic priority policies allow the priorities of queues to vary with respect to system parameters. Two dynamic priority schemes are presented: minimum laxity and queue threshold.

The minimum laxity (ML) policy [36, 40, 42, 44, 45] transmits cells that have the least amount of laxity. In this policy, the duration a cell can stay in a queue is pre-defined, and the count-down process starts as soon as a cell arrives in a buffer. Laxity is defined as the amount of time that remains before a cell's delay equals to the pre-defined duration. The ML policy is used mainly for delay-sensitive connections. This policy is also known as the *earliest due date* (EDD) policy.

To use ML to meet the end-to-end delay requirement in ATM networks, it is necessary to have information such as the time when a cell was transmitted from the source and the expiration time at each local node. These two pieces of information are not present in the current cell specifications, and pose difficulties in ML's implementation. Modifications to the parameters used in the ML are necessary before applying it to ATM nodes.

The queue threshold policy [38, 40, 41, 46] changes the priorities of different connections based on the available buffer space. The policy for two classes of traffic is illustrated in Figure 3.3. Class a is a high priority connection and class b a low priority one. Initially, class b is assigned a threshold. If the class b buffer is filled beyond the threshold, the priority for the class b traffic is changed from low to high, while the class a traffic is downgraded to the low priority. Once the class b buffer is reduced to less than the threshold, the priorities of the two classes of traffic are reverted to the original set-up.

Other variations are possible. For example, instead of downgrading the high priority connection, a round-robin policy may be used so both connections share the resources equally.



# of class b cells in queue	Priority	
	Class a	Class b
$\leq N_b$	High	Low
$> N_b$	Low	High

Figure 3.3 Queue Threshold Policy

Interleaving

The interleaving policy dictates the allocation of the transmitter by scheduling each connection in a pre-determined sequence. Each connection is scanned according to the sequence. If a connection does not have any information to send when its turn comes, it forfeits the allocation and the next connection on the list is scanned. If a connection has at least a cell in queue, it is allowed to send a cell. When the sequence is completed, it is repeated. A connection may be scanned multiple times during a sequence; in this case, the connection has been allocated more resources than other connections. The fairness issue usually appears during the analysis of an interleaving policy. The issue of fairness among connections and three interleaving schemes (round-robin, golden ratio, and virtual clock) are presented.

Fairness [44, 47-49] in sharing the transmitter resource has been an important issue since the packet-switched network era. In a packet network, fairness is critical because packets are allowed to have variable lengths, and long packets, by default, require more transmitter time for transmissions. However, since ATM networks use fixed-length cells, many complications no longer exist. Nevertheless, fairness in sharing the resource is still an important issues in ATM networks.

A popular definition of fairness is the *max-min fairness criterion*, which is stated as follows [44]:

1. No user receives more than its request,
2. No other allocation scheme which satisfies condition 1 has a higher minimum allocation, and
3. Condition 2 remains recursively true as the minimal user is removed and the total resource is reduced accordingly.

The mathematical version of the above conditions is

$$u_l = \text{MIN} (u_{fair}, \rho_l) \quad (3.1) \quad \text{condition 1}$$

$$u_{total} \leftarrow u_{total} - u_{min} \quad (3.2) \quad \text{condition 3}$$

$$u_{total} \geq \sum_{l=1}^N u_l \quad (3.3)$$

where

u_l = the resource allocated to user l

u_{fair} = the fair share, its value is calculated by dividing the remaining resource by the remaining number of users

ρ_l = the amount of resource requested by user l

u_{min} = the latest resource allocated

u_{total} = total resource available

N = total number of users

The max-min criterion achieves fairness in the following manner: the fair share (u_{fair}) is calculated by dividing the total resource among N users. If the minimal request is less than the fair share, the user is granted the amount of the resource requested, the total resource is reduced by the granted amount, and the fair share is re-calculated among $N-1$ users with the remaining resource. If the minimal request at any time is greater than the fair share, the resource is divided equally among the remaining users.

An implicit assumption made by the max-min fairness criterion is that all users have an equal priority to the resource. This assumption is not true within the ATM networks. Thus modifications are required to allow for traffic with different priorities.

The round-robin policy has been a popular scheme due to its fairness and ease of implementation [48, 50-52]. In [48], the fairness of the pure round-robin policy under the max-min fairness criterion is studied extensively. However, as pointed out earlier, the max-min criterion may not work well in a multi-priority environment. In response to the drawbacks, variants of pure round-robin policies have been proposed. Reference [52] presented a simulation study of the performance of the round-robin and its variants versus the first-come-first-served policy in the Xunet 2 environment. In this section, four schemes are presented: pure round-robin, weighted round-robin, prioritized round-robin, and framed round-robin.

The pure round-robin (PRR) policy scans each connection in turn, and transmits one cell if a connection has at least a cell waiting. If a connection does not need the service, it forfeits its allocated resource. The scanning continues until all the connections are visited once. The sequence starts again from the beginning of the sequence after its end is reached. The pure round-robin policy is extremely simple; however, it does not account for different transmission rates of connections, thereby becoming unfair to connections with high transmission rates.

Due to the different bandwidth requirements of different classes of connections, the weighted round-robin (WRR) policy is devised so that connections with large transmission rates can be accommodated. The basic policy is similar to the pure round-robin policy, except that more than one visit to a connection is allowed within a sequence. The multiple visits essentially increase the bandwidth allocation of the transmitter to a connection.

There are many ways to define a sequence for a WRR policy. The goal is to spread out the visits to a connection evenly to reduce jitter. An example is given here. Let a WRR sequence consist of multiple rounds. The number of rounds depends on the number of visits required. The number of visits, in turn, is determined by the relative magnitude of transmission rates among the connections. For instance, assume there are only two connections in the system with connection 1's transmission rate being 10 Mbps, and connection 2's rate being 5 Mbps. Then connection 1 is visited twice for every visit to connection 2, and there are two rounds in the sequence. The first round involves visits to connection 1 and 2, while the second round involves a visit to connection 2 only. Figure 3.4 depicts another example of a six-connection system, with two rounds per sequence. In this case, among the six connections, connections 1, 2, 4, and 5 are allocated twice as much bandwidth as connections 3 and 6. Reference [51] presents a scheduling policy for the different rounds to ease the delay jitters problem. Reference [50] presents a hardware implementation of WRR.

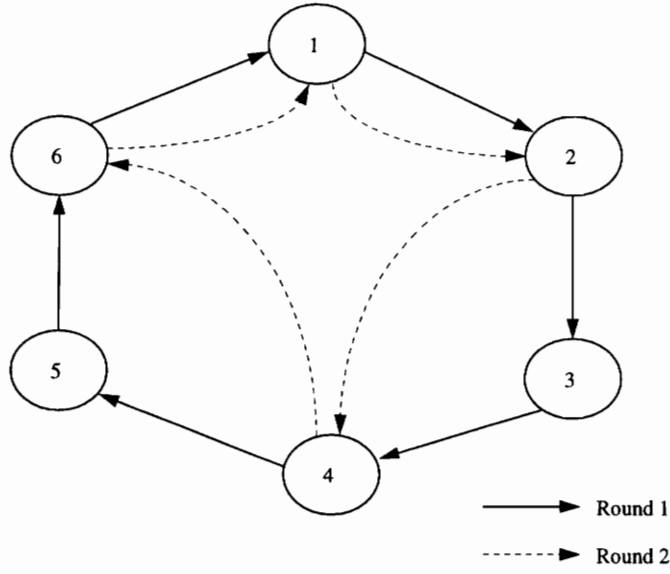


Figure 3.4 Weighted Round-Robin Policy

Prioritized round-robin and framed round-robin policies were studied via simulation in reference [52].

The prioritized round-robin policy is implemented in the following manner: Connections are divided into multiple lists based on their priorities. Within each list, the pure round-robin policy is used. The service moves from a higher priority list to a non-empty lower priority list if no more connections in the higher priority list requires transmission. The framed round-robin policy is implemented in the following manner: Connections are divided into two lists, list 0 and list 1. The transmission is divided into two phases, phase 0 and phase 1. When the system is in the phase 0, new connections are placed in the list 0, and connections in the list 1 are allowed to transmit in a pure round-robin manner; when the system is in the phase 1, connections are placed in the list 1, and connections from the list 0 are transmitted.

The golden ratio [53] interleaving policy utilizes the golden ratio, ϕ^{-1} , to schedule the transmission sequence of multiple connections. The constant ϕ is defined as

$$\phi^{-1} = \frac{\sqrt{5}-1}{2} \cong 0.618033989$$

and is related to the famous Fibonacci numbers via

$$F_k = \frac{\phi^k - (1-\phi)^k}{\sqrt{5}}$$

where F_k is defined by $F_{k+1} = F_k + F_{k-1}$ with $F_0 = 0$ and $F_1 = 1$.

Given S connections with requirements of X_l transmissions per sequence for connection l , $l = 1, 2, \dots, S$, and the total number of transmissions per sequence of N , $N = \sum_{l=1}^S X_l$, assign a constant a to each transmission, where $a = 1, 2, \dots, N$, in the following manner: for connection 1, the a 's assigned are $1, 2, \dots, X_1$, for connection 2, the a 's assigned are $X_1+1, X_1+2, \dots, X_2$. This assignment process continues until the last connection is assigned constants $X_{S-1}+1, X_{S-1}+2, \dots, N$. The actual transmission sequence is then determined by comparing values of modulus one of the products of a 's and ϕ^{-1} , with the smaller value transmit first.

For example, let $S = 4$, $N = 8$, $X_1 = 3$, $X_2 = 3$, and $X_3 = 2$. The sequence is calculated in Table 3.1. Thus, the transmission sequence is 2 1 3 2 1 2 1 3.

Table 3.1 The Calculations of Transmission Sequence for Golden Ratio Policy

# of Transmissions Requested	Constant a	$a \times \phi^{-1}$	$(a \times \phi^{-1}) \text{ MOD } 1$	Transmission Sequence
$X_1 = 3$	1	0.61803399	0.61803399	5
	2	1.23606798	0.23606798	2
	3	1.85410197	0.85410197	7
$X_2 = 3$	4	2.47213596	0.47213596	4
	5	3.09016994	0.09016994	1
	6	3.70820393	0.70820393	6
$X_3 = 2$	7	4.32623792	0.32623792	3
	8	4.94427191	0.94427191	8

The virtual clock policy was first proposed by Zhang [54] for packet-switched network. An ATM network implementation was suggested by Chao [55]. The goal of the virtual clock policy is to perform the bandwidth guarantees, protections of TDM systems, and the statistical multiplexing of packet-switched systems in a single policy. Actually, some of the components proposed in the virtual clock policy function

are similar to GCRA(I, L). A detailed description of the virtual clock policy is presented in the next few paragraphs.

The algorithm is divided into two parts: data forwarding and flow monitoring. Two global control variables are used in these two parts: a VirtualClock and an auxiliary VirtualClock (*auxVC*). In addition, the values of average rate (AR) and the average interval (AI) have to be supplied by users. The formal algorithm is presented below.

Data Forwarding

The switch uses output buffering implementation for each outgoing link. The queues serve packets in the following manner:

1. The first packet from $flow_i$ initiates the parameters for connection i .

$$VirtualClock_i \leftarrow auxVC_i \leftarrow \text{real time.}$$

2. Upon receiving each packet from $flow_i$,

$$(a) \ auxVC_i \leftarrow \text{MAX}(\text{real time}, auxVC_i)$$

$$(b) \ VirtualClock_i \leftarrow (VirtualClock_i + Vtick_i) \text{ and}$$

$$auxVC_i \leftarrow (auxVC_i + Vtick_i)$$

$$\text{If all packets have a constant size, } Vtick_i = \frac{1}{AR_i}; \text{ otherwise, } Vtick_i \text{ has}$$

to be calculated from individual packet size.

$$(c) \text{ Stamp the packet with the value of } auxVC_i$$

3. Insert the packet into its corresponding outgoing queue. The order of packets in the queue is in the increasing value of the time stamp.

Flow Monitoring

At the setup for $flow_i$, a control variable $AIR_i = AR_i \times AI_i$ is calculated. Based on the AIR_i , the $flow_i$ is monitored as follows:

1. If $(VirtualClock_i - \text{real time}) > T$, where T is a control threshold, then source i is transmitting too fast and flow control should be activated.

2. If ($VirtualClock_i < \text{real time}$), $VirtualClock_i \leftarrow \text{real time}$.

The transmission sequence is determined by the data forwarding of the virtual clock policy. The order is arranged based on a time stamp calculated with the average transmission rate and the size of packets. The flow monitoring part of the policy is used to build firewall protection of bandwidth, thus making sure that no source is sending information at a faster rate than the one agreed on.

The virtual clock policy is similar to GCRA(I, L) in its definition of parameters. The usage of $Vtick$ is similar to the parameter I. The purpose of the threshold control T is the same as the parameter L. The manner in which $VirtualClock$ is updated resembles the updating mechanism of the Theoretical Arrival Time.

3.2.3. Integrated Approach

The integrated approach includes algorithms that are designed to manage entire queue system. These algorithms may involve a combination of policies discussed in the previous sections, or new metrics are developed to perform discarding and transmission selections. Three policies are reviewed in this section: cost-based scheduling, head-of-line priority jump, and queue manager.

The cost-based scheduling policy [42] tries to optimize the system performance by first unifying different sets of performance parameters into a single metric through the use of cost functions. The cost functions are defined according to applications' performance objectives. For example, for real-time applications like voice and video, a cell that arrives too late represents no value to the end-users, and a cost incurred. At the same time, these applications tend to have high tolerance toward loss rate, thus a cell lost will generate a much smaller cost. Any action or inaction has a cost associated with it, and the decision of taking or not taking an action depends on the relative costs of all connections. Thus, if the cost incurred by withholding a cell from connection 1 and transmitting a cell from connection 2 is less than the cost incurred by transmitting a cell from connection 1 and withholding a cell from connection 2, then the cell from connection 1 is selected for transmission, while the cell from connection has to wait.

The objectives of the cost functions used in [42] are loss rate, mean delay, and delay allowance. For each objective, three functions are defined: the cost function, $c(t)$; the transmission priority function $p(t)$; and the discarding priority function, $d(t)$. The derivations of the mathematical representation for the three components are presented below.

Given $c_i(t)$, the cost incurred if packet i experiences a delay of t , the transmission priority and discarding priority functions for packet i by time t are

$$p_i(t) = \int_0^{\infty} a(\tau) c_i(q_{i,t} + \tau) d\tau - c_i(q_{i,t})$$

$$d_i(t) = \lim_{\tau \rightarrow \infty} c_i(\tau) - \int_0^{\infty} \frac{1}{\alpha} e^{-\tau/\alpha} c_i(q_{i,t} + \tau) d\tau$$

where

$a(\tau)$ = the distribution of the anticipated delay

$q_{i,t}$ = queuing delay packet i has experienced at time t

Loss Rate

If a cell is lost due to delay, the cost function would be a step function. Thus,

$$c(t) = \begin{cases} 0 & t \leq A \\ C & t > A \end{cases}$$

$$p(t) = \begin{cases} Ce^{-(A-t)/\alpha} & t \leq A \\ 0 & t > A \end{cases}$$

$$d(t) = \begin{cases} C(1 - e^{-(A-t)/\alpha}) & t \leq A \\ 0 & t > A \end{cases}$$

where

A = the tolerance of delay a cell can sustain in a queue

C = cost of a cell loss

α = the average value of the anticipated delay

Mean Delay

The cost function increases linearly with respect to the mean delay; however, the cost cannot increase indefinitely. A deadline, D , is introduced beyond which the cost of the mean delay stays constant. D is normally greater than A . Thus,

$$\begin{aligned}
c(t) &= \begin{cases} mt & t \leq D \\ mD & t > D \end{cases} \\
p(t) &= \begin{cases} m\alpha(1 - e^{-(D-t)/\alpha}) & t \leq D \\ 0 & t > D \end{cases} \\
d(t) &= \begin{cases} m(D-t) - m\alpha(1 - e^{-(D-t)/\alpha}) & t \leq D \\ 0 & t > D \end{cases}
\end{aligned}$$

where

D = deadline after which the cost of a cell becomes constant

m = cost per time unit

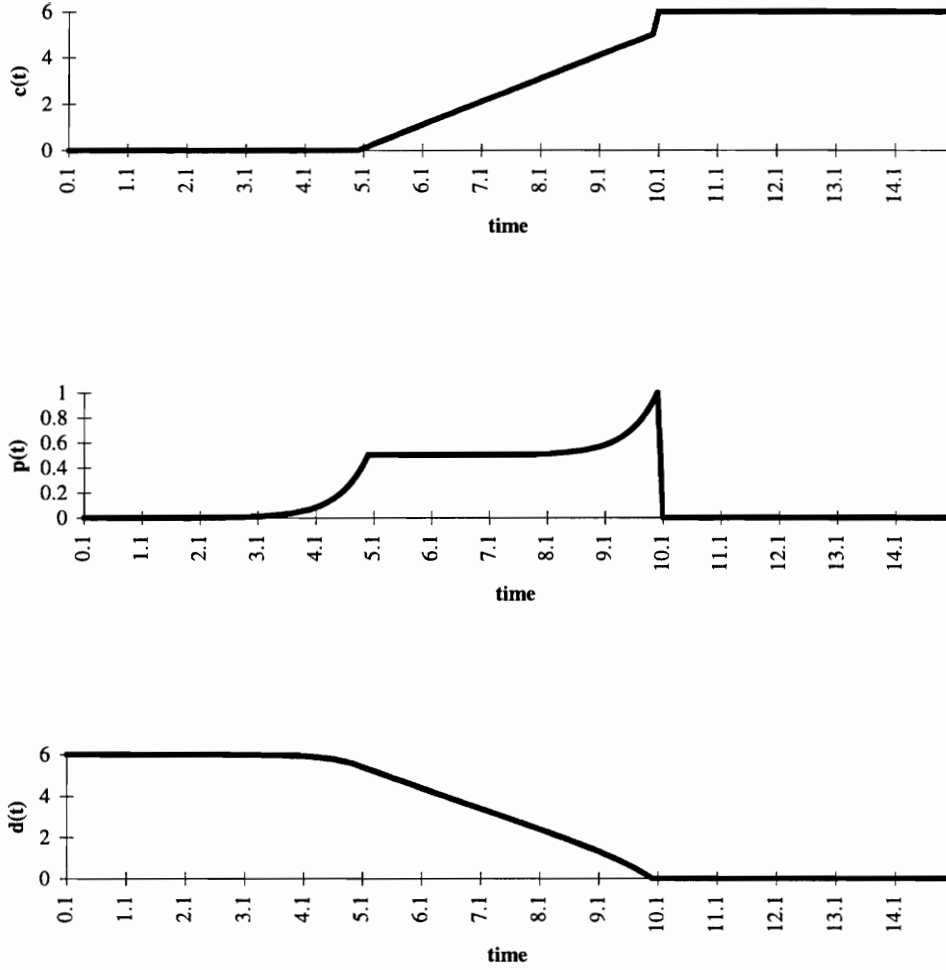
Delay Allowance

The delay allowance component is designed for applications for which certain amount of delay can be allowed without incurring costs, and incurring costs after the allowance expires. The cost function is then a ramp function. Thus,

$$\begin{aligned}
c(t) &= \begin{cases} 0 & t \leq A \\ m(t - A) & t > A \end{cases} \\
p(t) &= \begin{cases} m\alpha e^{-(A-t)/\alpha} & t \leq A \\ m\alpha & t > A \end{cases} \\
d(t) &= \lim_{\tau \rightarrow \infty} m\tau
\end{aligned}$$

Combining the three components, the resulting cost function and priority functions are

$$\begin{aligned}
c(t) &= \begin{cases} 0 & t \leq A \\ m(t - A) & A < t \leq D \\ m(D - A) + C & t > D \end{cases} \\
p(t) &= \begin{cases} m\alpha e^{-(A-t)/\alpha} + (C - m\alpha)e^{-(D-t)/\alpha} & t \leq A \\ m\alpha + (C - m\alpha)e^{-(D-t)/\alpha} & A < t \leq D \\ 0 & t > D \end{cases} \\
d(t) &= \begin{cases} m(D - A) + C - m\alpha e^{-(A-t)/\alpha} - (C - m\alpha)e^{-(D-t)/\alpha} & t \leq A \\ m(D - t) + (C - m\alpha)(1 - e^{-(D-t)/\alpha}) & A < t \leq D \\ 0 & t > D \end{cases}
\end{aligned}$$



$$A=5, D=10, m=1, C=1, \alpha=0.5$$

Figure 3.5 Cost-Based Scheduling Policy

Figure 3.5 illustrates the cost, scheduling and discarding priority functions for the objectives of mean delay, delay allowance, and loss rate.

The head-of-line priority jump (HOL-PJ) policy was proposed and analyzed under a Poisson arrival distribution in reference [45]. The goal of the policy is to bring fairness among different classes of traffic by making the tail probabilities of delay distributions for all classes approximately the same. According to [45], the queuing behavior of the HOL-PJ is similar to the earliest-due-date (EDD) discipline with no

preemption. In the essence, the HOL-PJ policy is a combination of the head-of-line queuing policy and dynamic priority policy. The policy is originally proposed for packet-switched networks, and a variant for ATM network is proposed by Cheng and Yegani in [38]. A detailed description of the HOL-PJ is presented in the next paragraph.

For each class (priority) of traffic, there is a delay limit on the maximum queuing delay for each packet. The limit is calculated as the difference of the delay requirement of the current class with that of the next higher priority class. If a packet stays in a queue for longer than the limit set for the queue, the packet's priority is changed to the next higher priority and is sent to the tail of the next higher priority queue, except for the highest priority queue. If a packet is sent to a higher priority queue, it is treated as a newly arrived packet and the delay limit of the new queue is applied to the packet. This jumping process continues until a packet reaches the highest priority queue, where the FCFS policy is implemented. The transmission policy is strictly a static priority, that is, the non-empty highest priority queue is served before a lower priority queue can be served. This operation is depicted in Figure 3.6.

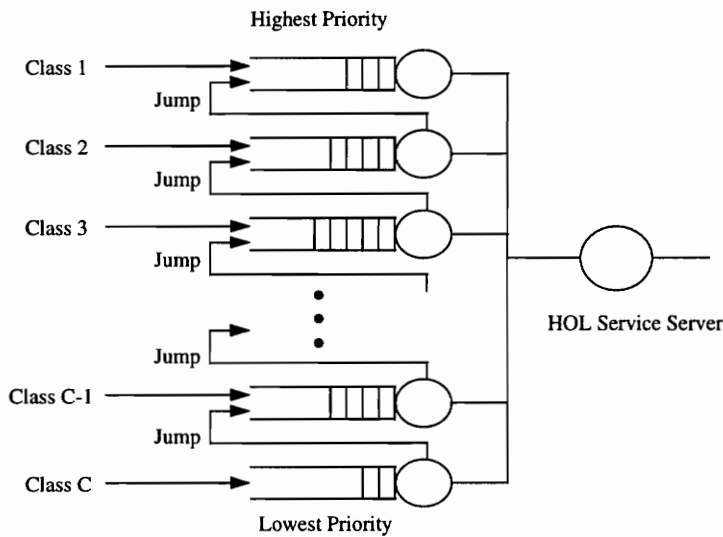


Figure 3.6 The Head-of-Line Priority Jump Policy

The queue manager (QM) scheme was proposed by Chao and Uzun [43] in 1992, and later extended by Chao and Pekcan [56] in 1994. In essence, the queue manager policy decides the cell discarding and transmission policy based on the loss priorities and delay priorities for all connections. To perform this task, a priority matrix is used. The matrix is formed by dividing connections into different classes by their loss and delay priorities. Based on the matrix, a discarding precedence and a transmission precedence can be

formed. These two precedence relationships govern the behavior of queues. Figure 3.7 depicts the queue manager for four classes of L connections.

According to Figure 3.7, there are L incoming connections which are divided into four classes based on their delay and loss priorities. There is a buffer pool, which dynamically allocates space for each class as cells arrive, and de-allocates space as cells leave. If the buffer pool is full, then the push-out policy takes effect, with the lowest priority cells pushed out first. The transmission sequence is performed based on the static priority scheme.

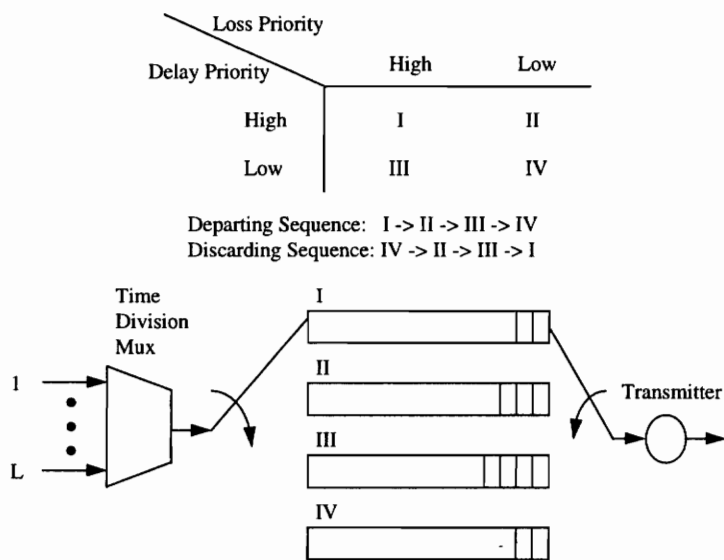


Figure 3.7 Static Priority Policy with Loss and Delay Priorities

Reference [56] extends this analysis to the following discarding and transmission precedence:

Transmission Precedence: (I = II) → (III = IV)

Discarding Precedence: (IV = II) → (III = I)

where

$a \rightarrow b$ indicates a has priority over b

$a = b$ indicates a and b has a equal precedence

3.2.4. Miscellaneous Policy

A framing strategy [32, 57] is presented in this section. The framing strategy was originally proposed for packet-switched network, and can be applied to the ATM network with slight modifications. The strategy is a combination of a throttle control scheme and a transmission policy, based on the idea of time-frames. A time-frame (or simply, frame) is defined as the duration of fixed-length T (not the peak emission interval), which divides the time-axis into multiple periods. The purpose of the framing strategy is to remove the possibility of congestion within the ATM network due to the aggregate behavior of all traffic. There are two parts to the strategy: packet admission policy and stop-and-go queuing.

The packet admission policy states that in order for a packet to be admitted, it has to be (r, T) -smooth. A stream of packets is (r, T) -smooth if during each frame of length T , at most $(r \cdot T)$ bits are transmitted, where r is the rate of transmission. If r is expressed in unit of packets per second (or cells per second in the case of ATM network), then the frames are bounded by $(r \cdot T)$ packets (cells) per second. Thus, given a bandwidth allocation of r_k for the connection k , the connection k 's rate of transmission into networks is bounded by $(r_k \cdot T)$ packets.

As pointed out in [32] and [57], rate throttle control does not guarantee that the congestion will not occur. Packets from different sources tend to cluster together as they proceed to the next node. The purpose of the stop-and-go queuing policy is to stop the clustering from appearing by enforcing the (r, T) -smooth property. The stop-and-go queuing policy can be stated as two rules:

Rule a. Given a node n , an incoming link l' , an outgoing link l , and arrival of a packet in the frame F' from link l' that needs to move to link l , a departing frame F of link l adjacent to the arriving frame F' , then the packet cannot be transmitted until the beginning of the frame F .

Rule b. A link should not be idle if the queue is non-empty.

Figure 3.8 depicts the stop-and-go queuing policy. A configuration of two incoming links and an outgoing link is shown. The frames are mismatched to show a general case of timing mismatch among nodes. The small up-arrows within the frames indicate presence of packets.

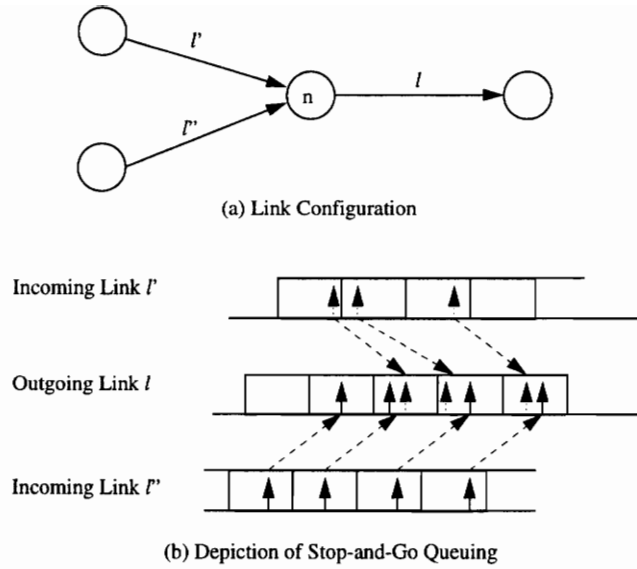


Figure 3.8 Stop-and-Go Queuing Policy

3.3. Summary

In this chapter, various policies regarding traffic and congestion control are reviewed. The traffic and congestion control techniques can generally be divided into two categories: rate throttle control and queue management. Rate throttle control schemes are interested in preventing congestion by regulating the rates at which sources can transmit. Queue management policies, on the other hand, deal with the discarding and transmission policies, which have strong impact on system behavior and performance. There have been many publications in this area, and this literature survey is by no means all inclusive.

CHAPTER 4. ALGORITHM DEVELOPMENT

In this chapter, a new algorithm is developed. The new algorithm is based on some policies proposed in the past but incorporates modifications in order to optimize the performance of ATM connections. The chapter is divided into six sections: Section 4.1 defines some terms to be used in conjunction with the new algorithm; Section 4.2 describes the buffer management policy; Section 4.3 presents the discard policy; Section 4.4 delineates the transmission policy; Section 4.5 presents an approximation approach to find the distributions of wait time and cdv-count (to be defined later); and Section 4.6 summarizes this chapter.

4.1. Definitions

The new algorithm consists of three separate policies, namely, buffer management, transmission, and discarding policies. In this section, a general description of the terms used in conjunction with the new algorithm is presented.

As indicated in Chapter 2, cell transfer delay (CTD), cell delay variation (CDV), and cell loss ratio (CLR) are the key parameters used in this research. However, in a traffic contract, these parameters are defined as end-to-end requirements, whereas the new algorithm is intended to be implemented in switching nodes and requires nodal information. In order to make these parameters relevant to a node, three parameters are introduced: the mean processing delay (MPD), the allocated-CDV (ACDV), and the allocated-CLR (ACLR). The MPD, ACDV, and ACLR are counterparts of the CTD, CDV, and CLR at the network level. The definitions of these three parameters are given in the following paragraphs.

The *mean processing delay* is the average processing delay allowed at each node. To find the MPD, two pieces of information are required: the end-to-end processing delay allowed, and the number of nodes which form the connection. As pointed out in Chapter 2, the CTD consists of two components, processing delay and propagation delay; the end-to-end processing delay can be found by subtracting the end-to-end delay requirement from the propagation delay. The number of nodes which form a connection, from source to destination, inclusively, is assumed to be known. The mean processing delay for a node is simply the result of dividing the end-to-end processing delay by the number of nodes. The MPD represents the amount of time that a cell can spend at each node and still arrive at the destination within the specified CTD.

The *allocated-CDV* is the amount of positive 1-point cell delay variations that each node can add to a cell, while allowing the cell's overall performance to meet the CDV requirement. Similar to the MPD, the ACDV can be found by dividing the CDV by the number of nodes that form a connection.

The *allocated-CLR* is the cell loss ratio that each node can have on average, while the actual end-to-end cell loss ratio still meets the requirement of the CLR specified in the traffic contract. The ACLR can be calculated by dividing the specified CLR by the number of nodes that form a connection.

The MPD, ACDV, and ACLR need to be calculated only once during a connection set-up. The implicit assumption made in their calculations is that all nodes within a connection have similar resources and traffic conditions. Nevertheless, since these values are used as references only, crude estimates are good enough. In addition, these parameters should be considered only as loose constraints, because it is possible for a cell to exceed its specified MPD, ACDV and ACLR at a congested node, and be compensated for at non-congested nodes, thereby still meeting the overall connection requirements.

4.2. The Buffer Management Policy

Most of the policies proposed in the literature assume a fixed-size buffer for each connection or priority class. As the buffer for a connection (or priority class) is filled up, cells are discarded according to a discard policy, even though other buffers may not be full. This approach under-utilizes precious system resources and increases the probability of cell loss. The buffer management approach of the Queue Manager proposed by Chao [43, 56] presented an interesting alternative and is adopted in the new algorithm.

The buffer management policy that is used treats all available buffer spaces as a single buffer pool. This buffer pool is then subdivided into cell spaces. A cell space consists of the actual cell and overhead information. The overhead information may include the cell arrival time (time stamp), connection identifier, sequence identifier, etc. The connection identifier and sequence identifier are used to maintain the queue structure of an individual connection. As cells arrive, cell spaces are allocated from the buffer pool. Once a cell is transmitted, the cell space is de-allocated and returned to the pool.

This buffer management policy provides two key advantages. First, the discarding of cells will not occur until all the available buffer spaces are used, thereby reducing the probability of cell loss for all connections. Second, the buffer resource is better utilized. However, the drawback is the complexity involved in the implementation of such a policy. For example, instead of a queue with an implicit arrival

sequence, identifiers are required to track the arrival sequence within each connection. Nevertheless, better performance for the policy may well be worth the complications.

The formulation of the buffer management policy is stated below.

Let a_i denote i cells arriving at a node from i different connections, and let b be the size of the buffer in terms of the unit of cell spaces, and b_{used} be the number of cell spaces in use. The buffer management policy is as follows:

If $(a_i + b_{used}) > b$,

Discard cells using a discard policy, (the discard policy used in the new algorithm is described in Section 4.3.) If any arriving cell remains after implementing the discard policy, allocate to it cell spaces from the buffer pool.

$$b_{used} = b_{used} + (\# \text{ of cells passed the discard policy})$$

If $(a_i + b_{used}) \leq b$,

Allocate cell spaces to the arriving cells,

$$b_{used} = b_{used} + i.$$

If a cell's transmission is completed,

De-allocate the cell space,

$$b_{used} = b_{used} - 1.$$

Since the buffer management policy used is the same as the one proposed by Chao's papers, readers are referred to [43, 56] for a complete analysis of this policy.

4.3. The Discard Policy

If the buffer becomes full, cells have to be discarded. The discard policy is used to select the cells to be discarded. In general, this selection is based on the static priorities of connections, or the order of cell arrivals, or both. A good discard policy for ATM should select cells with two objectives: (1). protect connections which are compliant with their traffic contracts, and (2). the actual cell loss ratios should remain within the specified ACLR over the long run for all connections.

Another issue that requires consideration is the interdependency of data. Information stored inside of consecutive cells may be related, and dropping a cell within a data burst may render the entire burst useless to the end-users. On the other hand, for certain applications, such as certain types of video, losing a cell within a burst does not significantly degrade the perceived quality; but losing multiple cells within a burst does cause serious degradation of service. Unless new parameters are adopted to indicate optimal ways for discarding cells of a connection (spread-out or concentrate), there is no easy way to distinguish between these two cases. Therefore, this issue is not considered in the discard policy of the new algorithm.

To achieve the two objectives stated earlier for the cell discard policy, a two-stage cell discard policy is proposed. The first stage involves the enforcement of a slightly modified version of Generic Cell Rate Algorithm (GCRA) to regulate each connection's incoming rate, and the second stage tries to select cells to be discarded based on rotation indices. These two stages are described in the following subsections.

4.3.1. Discard By MGCRA

To protect connections which are compliant with their respective traffic contracts, the first stage of the discard policy regulates the rates of the incoming cells from all connections. A slightly modified version of the GCRA (MGCRA) is used to check the incoming rates. The GCRA is discussed in Chapter 3, and is used in conjunction with two parameters, I and L . The MGCRA uses the value of peak emission period (T) for I , and the ACDV for L . The MGCRA differs from GCRA in the manner the rate regulation is activated. The GCRA is active at all times, checking incoming cells and discarding non-conforming ones. The MGCRA, on the other hand, is passive most of the time. If the buffer is not full, the MGCRA treats all cells as conforming cells, and updates the theoretical arrival time (TAT) for the next cell in a connection. If multiple cells from different connections arrive at any given time, and the available space in the buffer pool is not large enough to accept all the cells, then the MGCRA turns active by checking and discarding the arriving cells like a regular GCRA. Cells determined to be compliant are passed on to the second stage of the discard policy, while the non-conforming cells are discarded.

The MGCRA portion of the discard policy can now be formulated as follows.

Assume m cells arrive at time t from m different connections, and the number of cells that the buffer pool can admit is $(b - b_{\text{used}})$, where b is the size of the buffer (in cell spaces) and b_{used} is the number of cell spaces used. Let $TAT_{i,j}$ be the theoretical arrival time of the j^{th} cell from the i^{th} connection, $ACDV_i$ be the ACDV

value for the i^{th} connection, and T_i be the peak emission interval of the i^{th} connection. Also let i be the index defined over the m connections from which the cells arrive. Then,

0. Initialization. $TAT_{i,1} = 0, \forall i$

1. If $(b - b_{\text{used}}) \geq m$,

Then, if $TAT_{i,j} < t$, $TAT_{i,j+1} = t + T_i$, and accept the cell

Otherwise, $TAT_{i,j+1} = TAT_{i,j} + T_i$, and accept the cell

2. If $(b - b_{\text{used}}) < m$,

If $TAT_{i,j} < t$, then $TAT_{i,j+1} = t + T_i$, accept the cell

Otherwise,

If $(TAT_{i,j} - ACDV_i) > t$, and discard the cell

Otherwise, $TAT_{i,j+1} = TAT_{i,j} + T_i$, and accept the cell

4.3.2. Discard By Rotation

If the number of cells that pass the MGCRA is still larger than the number of cell spaces available in the buffer pool, then the second stage of the discard policy is activated; otherwise, all cells are accepted into the buffer pool. An index, the *rotation index*, is given to each connection based on the values of its allocated cell loss ratio (ACLR) and peak cell rate (PCR). The rotation indices indicate the number of cells to be dropped by each connection within a time period, such that the proportions of losses among connections are comparable to the proportions implied by the specified ACLR's in the long run. The values of the rotation indices are calculated as the product of ACLR and PCR for each connection; the reason for using the product of ACLR and PCR is given in the next paragraph. The connection with the smallest non-zero product of ACLR and PCR is given a rotation index of 1. The rotation indices of the other connections are calculated by first dividing the product of ACLR and PCR with the smallest non-zero product of ACLR and PCR, and then taking only the integer portion of the quotients. The cell to be dropped is the last arrived cell, including the cell arriving, from a connection with the largest rotation index. After a cell is dropped, the index for that connection is decremented by one. If there is no cell present for a connection when it is its turn to drop a cell, a cell from the connection with the next higher index is dropped. If multiple connections have the same index values, a cell from the connection with the largest ACLR is discarded. If a rotation

index has a value of zero and a cell from the corresponding connection needs to be dropped, the original rotation index values are added to the current values for all connections before the decrement, and the process continues.

The decision to use the product of a connection's ACLR and PCR values as a basis of the rotation index is based on the relationship of these products among the connections. Let a system with n connections have PCR's of $PCR_1, PCR_2, \dots, PCR_n$ and ACLR's of $ACLR_1, ACLR_2, \dots, ACLR_n$. From the definition of ACLR

$$\begin{aligned} ACLR_i &= \frac{\text{number of cells from connection } i \text{ that are discarded by the node}}{\text{total number of cells transmitted from connection } i \text{'s source}} \\ &= \lim_{\Delta t \rightarrow \infty} \frac{\text{number of cells from connection } i \text{ that are discarded by the node in } \Delta t}{\text{total number of cells transmitted from connection } i \text{'s source in } \Delta t} \\ &\text{where } \Delta t \text{ is a time period over which the ACLR's are defined.} \end{aligned}$$

Now, the total number of cells transmitted from connection i 's source during a time period of $\Delta t = PCR_i \times \Delta t$. Therefore, the ACLR becomes

$$ACLR_i = \lim_{\Delta t \rightarrow \infty} \frac{\text{number of cells from connection } i \text{ that are discarded by the node}}{PCR_i \times \Delta t}$$

If the limit notation is dropped, the resulting ratio is an approximation of the specified ACLR, or

$$ACLR_i \cong \frac{\text{number of cells from connection } i \text{ that are discarded by the node}}{PCR_i \times \Delta t}$$

The number of cells from connection i that are discarded by the node over a period of Δt is roughly $(ACLR_i \times PCR_i \times \Delta t)$. The ratio of the number of cells from connections i and j that are discarded is

$$\begin{aligned} &\frac{\text{number of cells from connection } i \text{ that are discarded by the node}}{\text{number of cells from connection } j \text{ that are discarded by the node}} \\ &= \frac{ACLR_i \times PCR_i \times \Delta t}{ACLR_j \times PCR_j \times \Delta t} \\ &= \frac{ACLR_i \times PCR_i}{ACLR_j \times PCR_j} \end{aligned} \quad (\text{Eqn. 4.1})$$

Thus, if the connection with the smallest non-zero product of ACLR and PCR is given a rotation index of 1, and all other connections are given the rotation indices according to Eqn. 4.1, using the ACLR and PCR from the connection which has a rotation index of 1 as the denominator, the second stage of the discard policy will force the pair-wise ratios of the actual ACLR's approaching the pair-wise ratios of the ACLR's specified in the traffic contract, as $\Delta t \rightarrow \infty$.

Note that the actual cell loss ratios at the node may or may not approach the specified ACLR's, depending on the size of buffer and the aggregate load on the transmitter. If a buffer is too small, and/or the load is too large, the actual cell loss ratios at the node will exceed the specified ones due to system's inability to cope with the load. On the other hand, if enough buffer space is available, the loss becomes small, and the actual cell loss ratios at the node is much less than the specified ACLR's. The variability in the actual cell loss ratios at the node in relation to the specified ACLR's makes it a poor candidate as a performance metric. The pairwise ratios of the actual cell loss ratios at the node, on the other hand, provides a much more stable relationship relative to the specified ACLR's, as the loss for all connections tend to increase and decrease at the same time. Thus, the pairwise ratios of actual cell loss ratio at a node of an ATM network is used to enforce the fairness of cell loss among the connections.

The formulation of the rotation portion of the discard policy is, then, as follows:

Step 0. Initialization. Let I_n be the rotation index for the n^{th} connection, $ACLR_n$ and PCR_n be the specified ACLR and PCR for the n^{th} connection, then

$$I_n = \begin{cases} 1, & \text{if } (ACLR_n \times PCR_n) = \min_{l=1, \dots, N} \{ (ACLR_l \times PCR_l) \} \\ \left\lfloor \frac{ACLR_n \times PCR_n}{\min_{l=1, \dots, N} \{ (ACLR_l \times PCR_l) \}} \right\rfloor, & \text{otherwise} \end{cases}$$

where $\lfloor \bullet \rfloor$ = the integer part of the value

Given the rotation indices for N connections, $I_1, I_2, \dots, I_N$, let $i_1 = I_1, i_2 = I_2, \dots, i_N = I_N$. Let $q_1, q_2, \dots, q_N$ be the number of cells in the connections. The connection from which the latest arrived cell is discarded is selected as follows:

Step 1. Let S be the set containing the non-empty connections with the largest rotation index values. Then,

$$S = \{s | i_s = \max_{n=1, \dots, N} \{i_n, q_n \neq 0\}\}$$

Step 2. Let W be the set containing the connections with the smallest ACLR's from the set S . Then,

$$W = \{w | ACLR_w = \min_{s \in S} \{ACLR_s\}\}$$

Step 3. Let l be the connection selected from W , or,

$$l = \text{RAND}(W), \quad \text{where RAND}(\bullet) \text{ is to randomly select from } \bullet.$$

Step 4. If $i_l = 0$, then $i_n = i_n + I_n$ for $n = l, \dots, N$

Step 5. Discard the cell from the connection selected. Also decrement the corresponding rotation index,

$$i_l = i_l - 1$$

No theoretical analysis is performed on the discard policy due to the tractability issues. Instead, the behavior of the discard policy is studied via simulation. The simulation based results of the discarding policy are presented in Chapter 5.

4.4. The Transmission Policy

As regards the transmission policy, the selection of packets to be transmitted is based on either the time of arrival, or the priority of the packet, or both. However, with ATM networks, these approaches ignore the fact that connections are allowed to have different requirements. In addition, a cell may have a time-varying priority. Time-varying priority is usually defined for a variable for which the priority increases or decreases as time progresses. Delay is a good example of time-varying priority. If a larger delay has higher priority, then as the time advances, the priority increases. As pointed out in [42], the delay of a piece of information in an integrated service has this characteristic.

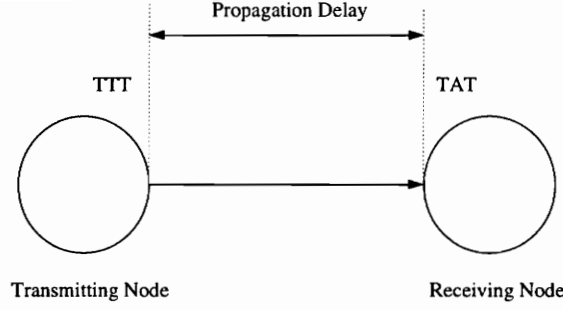


Figure 4.1 Locations Where the TTT and the TAT are Defined

Other concepts that need to be introduced are the *theoretical transmission time* (TTT), the *ACDV-window*, and the *MPD-window*. The TTT's definition is derived from the TAT, and the locations where the TTT and the TAT are defined is shown in Figure 4.1. For any two nodes, the TTT is defined for a cell leaving from the transmitting node, while the TAT is defined for a cell arriving at the receiving node. The TTT is offset from the TAT by a propagation delay of the link between these two nodes.

As referred to earlier, the TAT of a connection is used to estimate the arrival time of the next cell at a node once the current cell has been received. That is,

$$TAT_0 = a_0 = 0$$

$$TAT_{i+1} = \begin{cases} TAT_i + T & \text{if } TAT_i \geq a_i \\ a_i + T & \text{otherwise} \end{cases}$$

where

a_i is the actual arrival time of the i^{th} cell,

T is the peak emission interval, or $\frac{1}{\text{PCR}}$.

Note that the $(i+1)^{\text{st}}$ cell's TAT increases by an increment of T if the i^{th} cell arrives early or on time. If the i^{th} cell is late with respect to its TAT, the $(i+1)^{\text{st}}$ cell's TAT is the i^{th} cell's arrival time plus T . Also, another important fact is that the arrival time of the cells at the receiving node can be modeled as their transmission time at the transmitting node plus a propagation delay, because the cells' sequence do not change in a link and the propagation delays are the same for all the cells. Thus, in order to have the $(i+1)^{\text{st}}$ cell to arrive at

the receiving node exactly at TAT_{i+1} , given that the i^{th} cell is just transmitted from the transmitting node, the $(i+1)^{\text{st}}$ cell's transmission time should be T time units later. The theoretical transmission time can then be defined as T time units after the last transmission from a connection.

The ACDV-window is the period within which a cell may be transmitted from a node so that the end-to-end CDV is still bounded by the CDV specifications. The ACDV-window is defined with the TTT as the reference end point and $(TTT - ACDV)$ as the reference start point. However, if ACDV is greater than T , then the reference start point of the ACDV-window begins after the last transmission time rather than at $(TTT - ACDV)$. (Note that the TTT is defined as the last transmission time plus T .) Back-to-back transmission from a connection is allowed under some circumstances while still maintaining the end-to-end CDV bound. If the ACDV is smaller than T , then cells transmitted before the ACDV-window may or may not cause the bound on the overall CDV to fail, depending on how other nodes schedule the cell departures. Cells transmitted after the ACDV-window will not affect the CDV bound. Finally, the integer values of 1-point cdv a cell generates when it is transmitted will be called *cdv-count*. It is possible to have a negative 1-point cdv (or in discrete system, *cdv-count*), if the actual transmission time occurs after the TTT.

The MPD-window is a similar concept as the ACDV-window. The reference start point of a MPD window is a cell's arrival time, and the end reference point is MPD cell time units later.

The relationship among the TTT, the ACDV, the ACDV-window, the *cdv-count*, the MPD, and the MPD-window is illustrated in Figure 4.2. In Figure 4.2, the horizontal axis is the i^{th} cell's delay, and the vertical axis is the 1-point cdv of the i^{th} cell, if it is transmitted. The vertical axis is drawn at the transmission time of the $(i-1)^{\text{th}}$ cell. In this figure, the values used for T and ACDV are 6 and 4, respectively; thus, T cell time units (ctu's) away from the vertical axis is the TTT, and four ctu's prior to the TTT (or two ctu's after the vertical axis) is the beginning of the ACDV-window. If the ACDV is larger than 6, say, 8, the ACDV-window does not start from two cell time units before the vertical axis; rather, it begins at the vertical axis. The MPD-window starts at the i^{th} cell's arrival time, or when i^{th} cell's delay is zero, and ends at MPD ctu's later (4 ctu's after the vertical axis). If a cell is transmitted before the end of the MPD-window for all nodes that form the connection, the bound of CTD can be achieved. If a node is congested, it is possible for a cell's MPD-window and ACDV-window not to overlap, in which case, the cells are transmitted based on the relative urgency of various connections (the issue of relative urgency is addressed later). Note that if, in Figure 4.1, the $(i-1)^{\text{th}}$ cell is transmitted two cell time units after the end of MPD-window, then the vertical axis would be shifted to where the current TTT is located instead of what is shown, and the TTT and ACDV-window will be shifted accordingly. The dotted line, which starts at the 1-point cdv of 6 on the

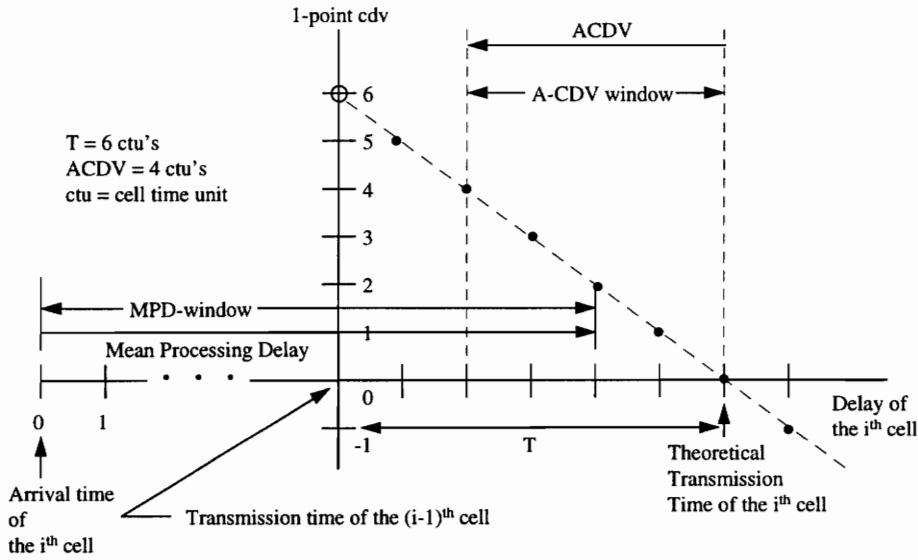


Figure 4.2 Trade-Off Between Delay and 1-Point CDV

vertical axis and intercepts the horizontal axis at the TTT, is the value of the 1-point cdv. The dots along the dotted line are the possible cdv-count values. Note that the 1-point cdv cannot be 6 (the empty circle on the vertical axis) because the 1-point cdv is defined with respect to the last transmission time; at the vertical axis, the last cell is being transmitted and the last transmission time is just determined.

When designing a transmission policy, the parameters and variables involved are the time-varying priorities associated with the delay and 1-point cdv. For a connection, after the transmission of the $(i-1)^{\text{th}}$ cell, it is desirable for the i^{th} cell to wait till its target transmission time (TTT). However, it is also preferred for the i^{th} cell to be transmitted as soon as possible to minimize delay. Thus, as the i^{th} cell waits in the queue, its delay approaches or exceeds the MPD, while the 1-point cdv is decreasing. It is desirable for a cell to be transmitted before the end of MPD-window and after the beginning of ACDV-window; that is, during the time period which MPD-window and the ACDV-window overlap. If the ACDV-window and the MPD-window do not overlap, a cell's optimal transmission time then has to be a compromise so that the combined effect of 1-point cdv and delay is minimized.

The proposed transmission policy uses a time-varying priority index scheme to determine the cell to be transmitted. Priority indices are calculated for cells waiting at the heads of connections by adding the scaled time-varying priority of delay and that of 1-point cdv, with a smaller index value indicates a higher priority. The time-varying priority of delay at time t is defined as the difference of t and the time at which the end of

the MPD-window occurs, and the time-varying priority of 1-point cdv is simply the value of 1-point cdv at t . These time-varying priorities are scaled by cost factors. The cost factors used are determined based on the relationship of a cell's delay and its time since the last transmission. For the time-varying priority of delay, if a cell's delay occurs within the MPD-window, then the cells' time-varying priority changes at a nominal rate (i.e. scaled by 1); otherwise, the rate of change of the cell's time-varying priority is scaled by a cost factor (c_1) that is greater than 1. The effect of scaling the rate of change with c_1 after the end of the MPD-window is to decrease the contribution to the priority index by the time-varying priority at a faster rate, since the time-varying priority becomes less than zero after the end of MPD-window (see Figure 4.3(a)). For the time-varying priority of 1-point cdv, if the cell's time since last transmission occurs before the beginning of the ACDV-window, a cost factor (c_2) that is greater than 1 is used to scale the rate of change of the time-varying priority; otherwise, a nominal cost factor of 1 is used. The effect of scaling the time-varying priority of 1-point cdv by c_2 is to increase its contribution to the priority index, therefore reducing the cell's priority to be transmitted (see Figure 4.3 (b)). In addition, different weighting factors can be used in conjunction with the scaled time-varying priorities to emphasize the relative importance of delay or 1-point cdv. Based on the priority indices of all connections, the connection with the smallest index value is selected to transmit a cell. The ACLR's are used only if there is a tie in indices: if multiple cells have the same priority index values, a connection with the smallest ACLR is chosen. If multiple cells have ties and the smallest ACLR value, then a connection is selected at random.

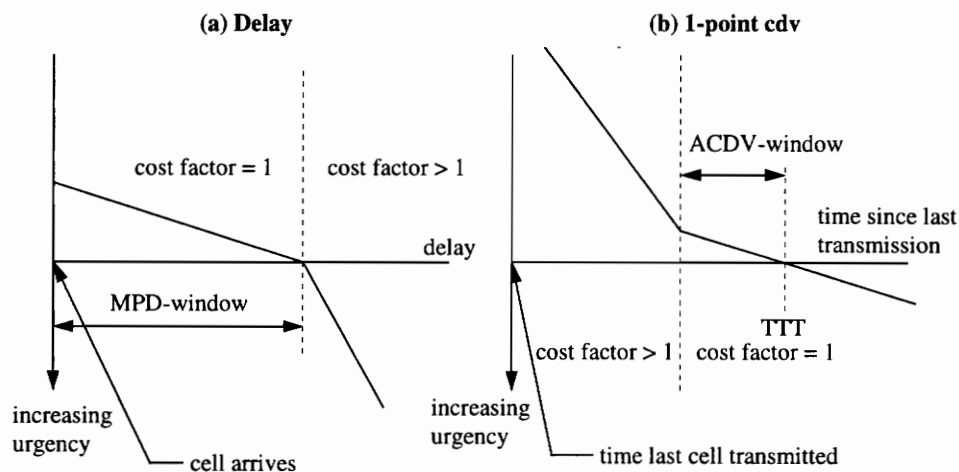


Figure 4.3 Generic Forms of the Scaled Time-Varying Priorities for Delay and 1-Point CDV

The formulation of the transmission policy is presented below.

Define the theoretical transmission time of the i^{th} cell in connection n , $TTT_{n,i}$, as follows:

$$TTT_{n,i} = \begin{cases} LT_n + \frac{1}{PCR_n}, & \text{if the current cell is not the first cell} \\ \text{current time}, & \text{otherwise} \end{cases}$$

where

LT_n = The last transmission time from connection n

PCR_n = The peak cell rate for connection n

Let C_n be the priority index for the cell at the head of connection n , v_n be the scaled time-varying priority of the cdv-count, and d_n be the scaled time-varying priority associated with delay, and α and β be the weights used to emphasize the relative importance of 1-point cdv and delay, respectively. Then

$$C_n = \alpha v_n + \beta d_n$$

$$d_n = \gamma(\text{time-varying priority of delay})$$

$$= \gamma(\text{MPD}_n - (\text{SYSCLK} - A_{n,i}))$$

where

$A_{n,i}$ = the arrival time of the i^{th} cell in connection n

SYSCLK = the current time in the system

$$\gamma = \begin{cases} 1, & \text{if } \text{SYSCLK} - A_{n,i} < \text{MPD}_n \\ c_1, & \text{if } \text{SYSCLK} - A_{n,i} \geq \text{MPD}_n \end{cases}$$

c_1 = cost factor applied for transmission after MPD-window ends, $c_1 > 1$

$$v_n = v_{n1} + c_2 v_{n2}$$

where

c_2 = cost factor applied for transmitting before the beginning of ACDV-window, $c_2 > 1$

v_{n1} = time-varying priority of the cdv-count after the ACDV-window begins for connection n

$$= \begin{cases} TTT_{n,i} - \text{SYSCLK}, & \text{if } \text{SYSCLK} \geq TTT_{n,i} - \text{ACDV}_n \\ \text{ACDV}_n, & \text{if } \text{SYSCLK} < TTT_{n,i} - \text{ACDV}_n \end{cases}$$

v_{n2} = time-varying priority of the cdv-count before the ACDV-window

$$= (TTT_{n,i} - ACDV_n - SYSCLK)^+$$

$$(\bullet)^+ \text{ is } \max(\bullet, 0)$$

Step 1. Let S_C be the set containing the subscripts of the connections with the smallest priority index. Or,

$$S_C = \{s | C_s = \min_{n=1, \dots, N} \{C_n\}\}$$

Step 2. Let X be the set containing the subscripts of the connections with the smallest ACLR and the smallest priority index. Then,

$$X = \{x | ACLR_x = \min_{i \in S_C} \{ACLR_i\}\}$$

Step 3. Let C be the subscript of the connection randomly selected from the set X . Thus,

$$C = \text{Rand}(X)$$

where

$$\text{Rand}(\bullet) = \text{random selection operation}$$

4.5. Approximation of the Wait Time and the cdv-Count Distributions

The distributions of the following three variables of the new algorithm are of interest: the delay, the 1-point cdv, and the loss ratio. As pointed out in Section 4.3, because of tractability issue, the loss distribution is studied via simulation. In this section, the derivation of an approximation method is presented. Because of certain assumptions made in the approximation method (to be listed later), instead of the distributions of the delay and the 1-point cdv, (which are continuous variables,) their discrete counterparts, the distributions of the wait time and the cdv-count, are estimated by an approximation method.

The wait time and the cdv-count distributions are the result of the transmission policy. A cell, that is at the head of a connection and has the smallest priority index value, is selected by the transmission policy; the priority index value is the weighted sum of the scaled time-varying priorities of the wait time and the cdv-count. The contribution from the wait time is denoted as d_n , and that from the cdv-count is denoted as v_n . Before the approximation approach can be presented, a different form of the priority index equation needs to be derived.

Priority Index Equation

Depending on the time at which the current cell's arrival and the previous cell's transmission take place, the priority index equation can be written in four different forms, with different combinations of cost factors used to scale the time-varying priorities. Figure 4.4. illustrates these four different conditions, labeled as R1, R2, R3, and R4. In condition R1, the current time is after the end of the MPD-window and after the beginning of the ACDV-window, thus the cost factor c_1 is used to scale the time-varying priority of delay, while a nominal cost factor of 1 is used to scale the time-varying priority of 1-point cdv. In condition R2, the current time is in the MPD-window and before the beginning of the ACDV-window, thus a nominal cost factor of 1 is used to scale the time-varying priority of delay, while the cost factor c_2 is used to scale the time-varying priority of 1-point cdv is low. In condition R3, the current time is in the overlap of the MPD-window and the ACDV-window, thus a nominal cost factor of 1 is used to scale both time-varying priorities. In condition R4, the current time occurs after the end of the MPD-window and before the beginning of the ACDV-window, thus the cost factor c_1 is used to scale the time-varying priority of delay, and the cost factor c_2 is used to scale the time-varying priority of 1-point cdv. Assuming that $\alpha = \beta = 1$, under the proposed formulation of the new algorithm, the priority index for connection n can be rewritten as

$$C_n = \begin{cases} [(TTT - SYSCLK)] + [(A + MPD_n - SYSCLK)] & \text{if in R3} \\ [(TTT - SYSCLK)] + [c_1 \cdot (A + MPD_n - SYSCLK)] & \text{if in R1} \\ [ACDV_n + c_2 \cdot (TTT - SYSCLK - ACDV_n)] + [(A + MPD_n - SYSCLK)] & \text{if in R2} \\ [ACDV_n + c_2 \cdot (TTT - SYSCLK - ACDV_n)] + [c_1 \cdot (A + MPD_n - SYSCLK)] & \text{if in R4} \end{cases} \quad (\text{Eqn. 4.2})$$

where A = the arrival time of the cell under consideration, or the tagged cell's arrival time

$SYSCLK$ = the current time

R1 is the region such that $SYSCLK \geq TTT - ACDV$ and $SYSCLK > A + MPD$

R2 is the region such that $SYSCLK < TTT - ACDV$ and $SYSCLK \leq A + MPD$

R3 is the region such that $TTT - ACDV \leq SYSCLK \leq A + MPD$

R4 is the region such that $A + MPD < SYSCLK < TTT - ACDV$

Some subscripts of the variables are dropped, since only one connection is under consideration, and the only cell involved is the cell at the head of the connection. It is not necessary to identify specific cells.

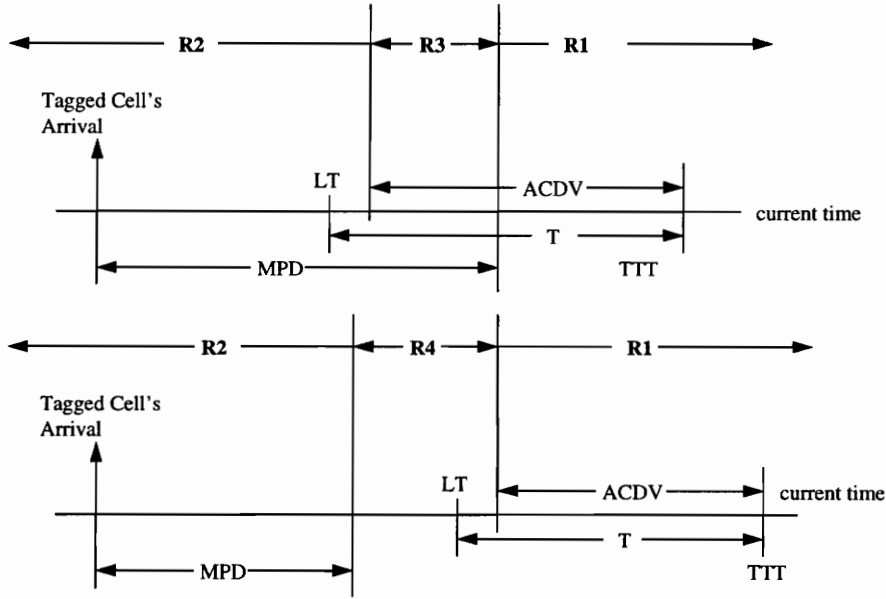


Figure 4.4 Possible Scenarios under the Proposed Algorithm

The first set of square brackets in each expression of Eqn. 4.2 is the time-varying priority of the cdv-count, while the second set of square brackets is that of the delay. Substituting $(LT + T)$ for TTT , and rearranging the terms, the following equation is obtained:

$$C_n = \begin{cases} [T_n - (SYSCLK - LT)] + [MPD_n - (SYSCLK - A)] & \text{if in R3} \\ [T_n - (SYSCLK - LT)] + [c_1 \cdot (MPD_n - (SYSCLK - A))] & \text{if in R1} \\ [ACDV_n + c_2 \cdot (T_n - (SYSCLK - LT) - ACDV_n)] + [MPD_n - (SYSCLK - A)] & \text{if in R2} \\ [ACDV_n + c_2 \cdot (T_n - (SYSCLK - LT) - ACDV_n)] + [c_1 \cdot (MPD_n - (SYSCLK - A))] & \text{if in R4} \end{cases}$$

The expression $(SYSCLK - LT)$ is the amount of time since the last transmission for the same connection, while $(SYSCLK - A)$ is the amount of time since the arrival of the tagged cell. These two quantities can be used to indicate a connection's status relative to other connections. To simplify the notation, two connection variables, a and lt , are introduced, where

$$\begin{aligned} a &= SYSCLK - A \\ lt &= SYSCLK - LT \end{aligned}$$

The variables a and lt are referred to as connection variables because at all times, only the cell at the head of each connection is considered. Thus, the transmission policy can select the connection from which a

transmission is made based on these two variables of all the connections. These two variables also indicate the evolution of a connection's status since the previous transmission (more details are given in the next sub-section).

Substituting the two new variables into the equations, and recognizing the conditions in which the equations are valid, the priority index equation can now be simplified as

$$C_n = (T_n - lt) + (c_2 - 1) (T_n - lt - ACDV_n)^+ + (MPD_n - a)^+ - c_1 (a - MPD_n)^+$$

Thus, the priority index of a cell at the head of connection n is expressed in terms of the connection's parameters (T , $ACDV$, and MPD) from the traffic contract, costs (c_1 and c_2), and two connection variables (a and lt).

Approximation of The Wait Time And The cdv-count Distributions

Two main obstacles need to be resolved while analyzing the transmission policy mathematically:

1. The transmission policy involves comparisons of several connections' priority indices, which are calculated based partially on the connections' characteristic parameters (T 's, $ACDV$'s, and MPD 's). Unless the values of parameters for all the connections are known, the transmission policy can not be defined. However, once the values of the parameters are set, the analysis is applicable to that set of values only.
2. The comparison of priority indices among all connections makes the analysis of a single connection impossible, since any decision made by the transmission policy affects all connections containing cells at that time. For example, the transmission policy's decision to select a cell from a connection implies other connections with cells have to wait. Thus, the wait time and the cdv-count distributions of all connections need to be determined simultaneously.

To resolve the first obstacle, this research does not intend to conduct studies for all combinations of possible parameter values; rather, an approach on how the system can be solved, once the parameter values are given, is considered. To overcome the second obstacle, a Markov chain method with certain assumptions can be used. However, the main difficulty with this approach is that the number of possible states of the Markov chain explodes even for a system with few connections. Nevertheless, an approximation of the Markov chain can be used and a solution can be found.

The following assumptions are made for the derivation of the wait time and cdv-count distributions.

1. Events, such as arrivals and transmissions, take place only at the beginning of discrete time slots.
2. The selection of connections to transmit cells is performed with the knowledge of new arrivals, even though these two events (transmission and arrivals) are assumed to occur at the same instant as indicated by Assumption 1.
3. The arrival processes of all the connections are Bernoulli, with the probabilities of success being equal to the ratios of connections' peak cell rates to the transmitter's transmission rate.
4. The overall system is stable, i.e. the long run average load is less than the server capability.
5. The buffer size is infinite.
6. A server can not be idle if cells are available.
7. No connection is added or removed from the system.
8. The system has only two connections.
9. If a connection is empty, then the cdv-count is assumed to be a infinitely large positive integer.

Let the ordered pair (a, lt) denote a tagged cell at the head of a tagged connection (or a tagged queue) with the variables a and lt . Thus a tagged cell with an ordered pair of $(2, 7)$ indicates that the previous cell from the tagged queue is transmitted seven cell time units earlier, and the current cell has been waiting in the queue for two cell time units. The past values for the tagged queue since the transmission of the previous cell are $(E, 1)$, $(E, 2)$, $(E, 3)$, $(E, 4)$, $(0, 5)$, and $(1, 6)$. The letter 'E' indicates that the tagged queue is empty at the lt cell time units since the previous cell is transmitted. The valid values for a are E and non-negative integers, while for lt they are positive integers only. The variable a can be zero because of the assumption

that the server can consider cells for transmission as they arrive during the transmission selection. The variable lt can not have a zero value because it takes one time unit before the next decision is made; thus, if a cell from a tagged connection is transmitted one time unit ago, the tagged queue would have $lt = 1$ for the current decision.

The current status of a connection can be represented by the (a, lt) pair and by the arrival pattern of the cells waiting in the buffer of that connection. The arrival pattern contains the information as to when each cell arrives relative to the cell before it. The arrival time offset is used to calculate the wait time of the next cell once the cell at the head of the connection's queue (current cell) is transmitted. For example, if for a current cell with $a = 3$, and the next cell arrives two cell time units after the current cell's arrival, and the current cell is transmitted now, then the next cell has a wait time of 2 ($= 3 - 2 + 1$, one cell time unit is added since the next decision is made a cell time unit later) when the transmission policy makes the next decision. The arrival pattern can be represented by a binary number. Each digit in a binary number can be used to indicate the presence of a cell, with the largest non-zero digit being the current cell. Using the example just given, and further assuming that only two cells from a connection are in the buffer, the arrival pattern and the corresponding binary number is shown in Figure 4.5. If the current cell is not transmitted and there is a cell arriving now, then the binary number at the next time slot becomes 10101. If the current cell is transmitted now and there is no cell arriving, then the binary number at the next time slot becomes 100. Let B denote the binary number which represents the arrival pattern in a connection, then, an ordered triple (a, lt, B) can be used to show the status of the entire connection.

A Markov state is defined as an ordered $3n$ -tuple, where n is the number of connections. The elements of

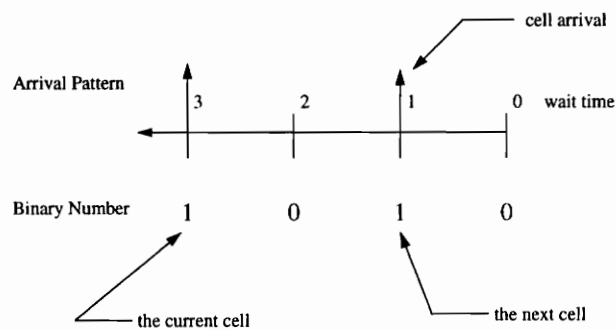


Figure 4.5 The Binary Representation of Arrival Pattern

the $3n$ -tuple are taken from the ordered triple, (a, lt, B) , of n connections. Thus, in a two-connection system, a Markov state is an ordered 6-tuple $(a_1, lt_1, B_1, a_2, lt_2, B_2)$. Given a Markov state at the beginning of a time slot, the priority indices for all connections are known, thus the connection selected to be transmitted is determined.

The evolution of the Markov chain depends on the connection that is selected to transmit a cell, and the arrivals from all the connections. Let $(a_{1,t}, lt_{1,t}, B_{1,t}, a_{2,t}, lt_{2,t}, B_{2,t})$ and $(a_{1,t+1}, lt_{1,t+1}, B_{1,t+1}, a_{2,t+1}, lt_{2,t+1}, B_{2,t+1})$ be the states of the Markov chain at times t and $t+1$, for the two-connection system. Without loss of generality, assume that the connection 1 in the two-connection system is selected for transmission at time t . The possible states that the system can be in at time $t+1$ given its state at time t depends upon the arrival of cells from all connections, and are as follows:

$$a_{1,(t+1)} = \begin{cases} E & \text{if } B_{1,(t+1)} = 0 \\ (B_{1,(t+1)})^{>>} & \text{if } B_{1,(t+1)} \neq 0 \end{cases} \quad (\text{Eqn. 4.3.1})$$

where

$$\begin{aligned} (\bullet)^{>>} &= \text{number of times } \bullet \text{ needs to be right-shifted to reach 1} \\ &= \text{the wait time of the cell at the most significant bit position} \end{aligned}$$

$$lt_{1,(t+1)} = 1 \quad (\text{Eqn. 4.3.2})$$

$$B_{1,(t+1)} = \begin{cases} 2 \cdot (B_{1,t})^* + 1 & \text{if an arrival occurs in connection 1} \\ 2 \cdot (B_{1,t})^* & \text{if no arrival occurs in connection 1} \end{cases} \quad (\text{Eqn. 4.3.3})$$

where

$$(\bullet)^* = \bullet \text{ with the largest non-zero digit set to zero}$$

$$a_{2,(t+1)} = \begin{cases} E & \text{if } a_{2,t} = E, \text{ and no arrival occurs in connection 2} \\ a_{2,t} + 1 & \text{if } a_{2,t} \neq 1, \text{ an arrival or no arrival occurs in connection 2} \\ 0 & \text{if } a_{2,t} = E, \text{ and an arrival occurs in connection 2} \end{cases} \quad (\text{Eqn. 4.3.4})$$

$$lt_{2,(t+1)} = lt_{2,t} + 1 \quad (\text{Eqn. 4.3.5})$$

$$B_{2,(t+1)} = \begin{cases} 2 \cdot B_{2,t} + 1 & \text{if an arrival occurs in connection 2} \\ 2 \cdot B_{2,t} & \text{if no arrival occurs in connection 2} \end{cases} \quad (\text{Eqn. 4.3.6})$$

Note that for a binary number, multiplying by two is equivalent to shifting the digits to the left by one digit, with the least significant digit becoming zero.

The effect of arrivals at the beginning of the next time slot on the evolution of the Markov chain is shown in Eqn. 4.3.1, Eqn. 4.3.3, Eqn. 4.3.4, and Eqn. 4.3.6. Eqn. 4.3.2 and Eqn. 4.3.4 are not affected by the arrival processes. From these equations, it is clear that the evolution of the Markov chain is driven by the arrival processes of the connections. Because of the Bernoulli assumption on the arrival processes and the slotted time implementation, there are four possible ways for the arrivals to occur in the presence of two connections: no arrival for both connections, only one arrival in connection 1, only one arrival in connection 2, and arrivals in both connections. The probabilities of these four combinations are $(1-\lambda_1)(1-\lambda_2)$, $\lambda_1(1-\lambda_2)$, $(1-\lambda_1)\lambda_2$, and $\lambda_1\lambda_2$, respectively, where λ_1 and λ_2 are the ratios of the peak cell rates of connection 1 and 2 over the transmitter's transmission rate. These probabilities are the transition probabilities for all states since the arrivals determine the next state. Figure 4.6 illustrates a generic Markov state with its transition probabilities.

The reason that the number of states explodes even for a small system becomes self-evident. For a two-connection system, the space generated by the Markov states has six dimensions. For every additional

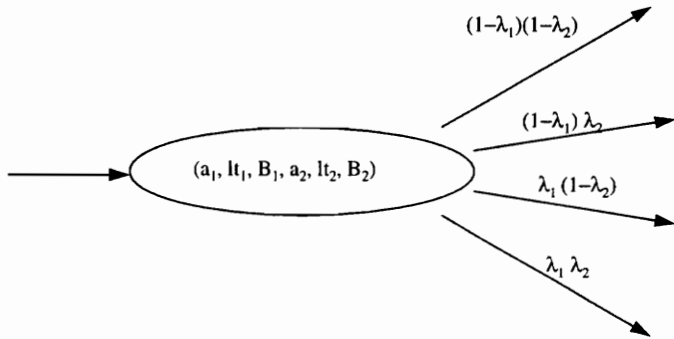


Figure 4.6 A Generic Markov State for a Two-Connection System

connection added to the system, the solution space grows by additional three dimensions. One way to control the number of states is by capping the largest values of a and lt . If the values of a and lt are limited, the resulting Markov chain will be a portion of the original Markov chain; hence the solutions found are only approximations.

To generate the approximated Markov chain, the following steps are used:

Step 1. Identify the search space for each connection. A search space is a two-dimensional plane with the valid values of a depicted on the vertical axis and that of lt on the horizontal axis, and is defined once the maximum values of a and lt are given. Using the parameters given in Table 4.1, and using the maximum a of 1 and lt of 2 for both connections, the search space for the two-connection system is illustrated in Figure 4.7. The dots in Figure 4.7 marks the possible a and lt values that a cell at the head of connection can have. The numbers beneath the dots are the corresponding priority index values. Thus, there are six valid points in the search space of each connection. Define a border point as the point having a maximum possible value for either its a or lt , and an interior point as a point that is not a border point. Thus, in Figure 4.7, the points (E, 2), (0, 2), (1, 1), and (1, 2) are border points for connection 1, and (E, 1), (0, 1) are interior points.

Table 4.1 Connection Parameters for a Two-Connection System Example

	λ	MPD	T	ACDV
Connection 1	0.1	10	5	1
Connection 2	0.1	10	5	1

$$c_1 = 2 \quad c_2 = 2$$

Step 2. Generate the candidate states. The candidate states are the permutations of the valid points in the search spaces of all connections. Thus, for the two-connection system defined above, there are thirty six candidate states; these states are listed in Table 4.2. States that are not candidate states are considered to be invalid, and are ignored. Note that the candidate states are not the Markov states. Candidate states contain only information on a and lt , while Markov states have a , lt , and the arrival pattern, B.

Step 3. Eliminate candidate states which violate the assumptions. Generally, there are the following four rules that can be used to eliminate the candidate states:

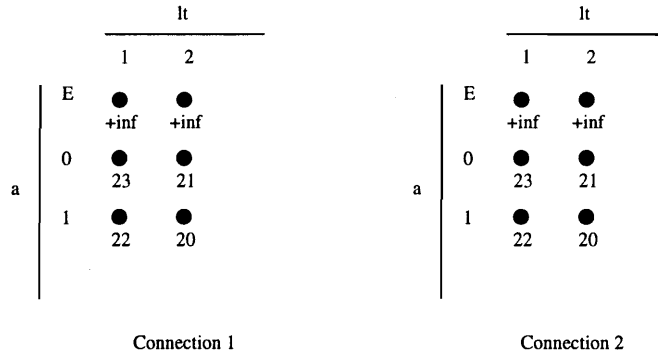


Figure 4.7 Possible States and Priority Indices in a Two-Connection Example

Table 4.2 Candidate States

Connection 2							
Connection 1		E, 1	E, 2	0, 1	0, 2	1, 1	1, 2
	E, 1	E,1,E,1	E,1,E,2	E,1,0,1	E,1,0,2	E,1,1,1	E,1,1,2
	E, 2	E,2,E,1	E,2,E,2	E,2,0,1	E,2,0,2	E,2,1,1	E,2,1,2
	0, 1	0,1,E,1	0,1,E,2	0,1,0,1	0,1,0,2	0,1,1,1	0,1,1,2
	0, 2	0,2,E,1	0,2,E,2	0,2,0,1	0,2,0,2	0,2,1,1	0,2,1,2
	1, 1	1,1,E,1	1,1,E,2	1,1,0,1	1,1,0,2	1,1,1,1	1,1,1,2
	1, 2	1,2,E,1	1,2,E,2	1,2,0,1	1,2,0,2	1,2,1,1	1,2,1,2

1. The transmitter can transmit only one cell at a time, which implies that the values of lt can not be the same for any pair of connections. Using this rule, candidate states (E,1,E,1), (E,1,0,1), (E,1,1,1), (E,2,E,2), (E,2,0,2), (E,2,1,2), (0,1,E,1), (0,1,0,1), (0,1,1,1), (0,2,E,2), (0,2,0,2), (0,2,1,2), (1,1,E,1), (1,1,0,1), (1,1,1,1), (1,2,E,2), (1,2,0,2), and (1,2,1,2) are eliminated.

2. If a candidate state is made up by more than one border points, the candidate state is eliminated. If a connection is represented by a border point, then the connection must be allowed to transmit by the next time slot, or the connection has to be represented by points outside of its search space and it is not allowed. When a candidate state is

consisted of two or more border points, since only one connection is allowed to transmit, the status of other connections will become invalid by the next time slot. For example, the candidate state (1,1,0,2) is made up by points (1,1) and (0,2) in the search spaces of connection 1 and 2, respectively. Since both (1,1) and (0,2) are border points in their respective search spaces, selecting one connection will force the other one to be an invalid point. With this rule, states (1,1,E,2), (1,1,0,2), (1,1,1,2), (E,2,1,1), (0,2,1,1), and (1,2,1,1) are eliminated.

3. No interior points in a search space should be compared with a border point from other search spaces with E as its a . The reason for this rule is similar to rule 2. If a candidate state is composed of valid interior points of search spaces and at least a border point with E as its a , one of the connections represented by the interior points will be selected to transmit a cell, and it will force the connection with the border point to become invalid. States eliminated under this rule are (E,1,E,2), (0,1,E,2), (E,2,E,1), and (E,2,0,1).

4. Transmitter is never idle. Thus, the lt 's of connections can not be all greater than 1 while at least one a is greater than zero. Since no state falls within the condition for this rule, the remaining states are valid candidate states.

The candidate states after applying the four rules are the valid candidate states. For the two-connection system, the valid candidate states are listed in Table 4.2 with shadows.

Step 4. Expand each valid candidate state by considering possible arrival patterns. Given the wait time of cells at the heads of the connections, all possible arrival patterns with the same a 's and lt 's can be generated. Thus, each valid candidate state (with four elements in the two-connection system) is expanded into multiple candidate Markov states (with six elements in the two-connections system) by attaching possible B's with decimal values ranging from zero to a .

Step 5. For each candidate Markov state, find the next four candidate Markov states based on Eqn. 4.3.1-4.1.6. If any of the next candidate Markov states is not among the candidate Markov states expanded in step 4, then the transitions leading to this state is invalid (as it evolves into an invalid Markov state) and is eliminated. The probabilities for the remaining transitions has to be adjusted accordingly, since the sum of all transition probabilities to the next states has to be one, and each candidate Markov state has to evolve to

a different candidate Markov state. For a two-connection system, the transition probabilities after removing the ones leading to invalid Markov states can be calculated as

$$P(b_1, b_2 \mid \text{the next state is valid}) = \frac{P(b_1, b_2 \mid \text{all next states are valid})}{1 - \sum_{c_1=0}^1 \sum_{c_2=0}^1 P(c_1, c_2 \mid \text{the next state is invalid})}$$

where

$P(b_1, b_2)$ is the probability that connection 1 has b_1 arrival and connection 2 has b_2 arrival
 $b_1, b_2 = 0 \text{ or } 1$

Figure 4.8 illustrates the process for the Markov state $(E, 1, 0, 0, 2, 1)$ in the two-connection system example. If there is no arrival for either connection by the next time slot, the next Markov state becomes $(E, 2, 0, E, 1, 0)$, which is not a valid state (see Table 4.2). Similarly, the next Markov state $(E, 2, 0, 0, 1, 1)$ is also an invalid state. Thus, these two transitions are removed and the remaining two transition probabilities are adjusted.

Step 6. Eliminate the remaining valid Markov states which do not have any transition probability. There can not be any terminating Markov state in the Markov chain. Any valid Markov state which does not have any

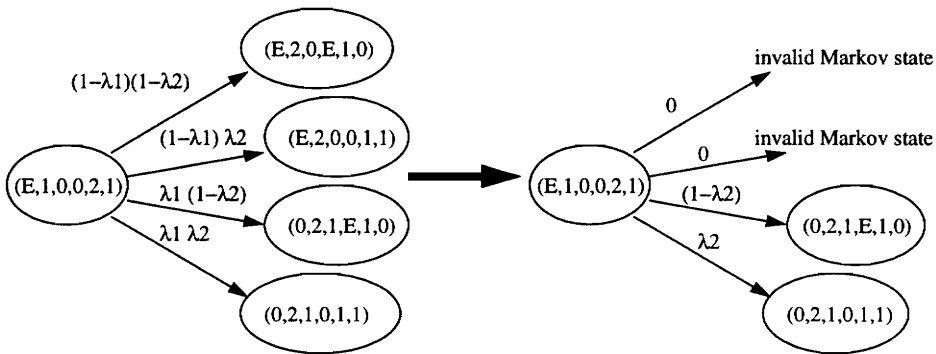


Figure 4.8 An Example of Step 5

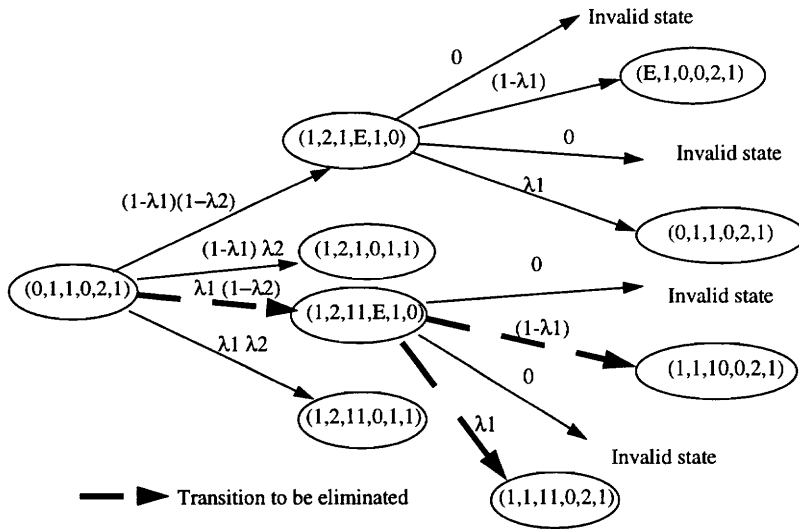


Figure 4.9 State Removal Due To No Transition Probability

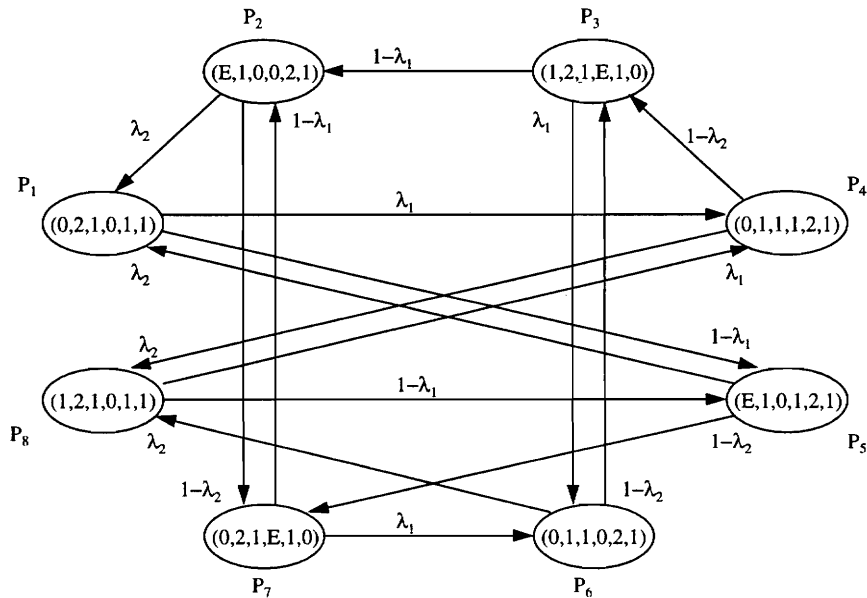


Figure 4.10 Markov Chain for the Two-Connection System Example

transition probability has to be removed, and the transition probabilities of other Markov states that evolve in this state has to be adjusted accordingly. An example is illustrated in Figure 4.9. In Figure 4.9, the Markov state (1,2,11,E,1,0) has no transitional probability, therefore it is eliminated. The transition of state (0,1,1,0,2,1) that evolves into (1,2,11,E,1,0) has to be set to 0, and the other transitional probabilities have to be adjusted accordingly (transitional probability adjustment not shown in Figure 4.9).

Based on the rules discussed so far, there are the following eight valid Markov states that remain in the two-connection system example: (0,2,1,0,1,1), (E,1,0,0,2,1), (1,2,1,E,1,0), (0,1,1,1,2,1), (E,1,0,1,2,1), (0,1,1,0,2,1), (0,2,1,E,1,0), and (1,2,1,0,1,1). The resulting Markov chain is shown in Figure 4.10.

Once the approximated Markov chain is developed, a system of equations can be written. For the Markov chain shown in Figure 4.10, the linear state transition equations are as follows:

$$\begin{aligned}
 P_1 &= \lambda_2 (P_2 + P_5) \\
 P_2 &= (1-\lambda_1) (P_3 + P_7) \\
 P_3 &= (1-\lambda_2) (P_4 + P_6) \\
 P_4 &= \lambda_1 (P_1 + P_8) \\
 P_5 &= (1-\lambda_1) (P_1 + P_8) \\
 P_6 &= \lambda_1 (P_3 + P_7) \\
 P_7 &= (1 - \lambda_2) (P_2 + P_5) \\
 P_8 &= \lambda_2 (P_4 + P_6)
 \end{aligned}
 \tag{Eqn. 4.4}$$

In addition, the sum of all probabilities must be one. Thus

$$P_1 + P_2 + P_3 + P_4 + P_5 + P_6 + P_7 + P_8 = 1 \tag{Eqn 4.5}$$

The system of linear equations (Eqn. 4.4 and Eqn. 4.5) can be solved readily. Using the connection parameters given in Table 4.2, the solution is tabulated in Table 4.3.

To find the distribution of wait time for a connection, the following steps are used:

1. Divide the states into groups based on the connection selected for transmission.

Table 4.3 State Solution to the Two-Connection System Example

	P ₁	P ₂	P ₃	P ₄	P ₅	P ₆	P ₇	P ₈
Probability	0.045	0.405	0.045	0.005	0.045	0.045	0.450	0.005
State	(0,2,1, 0,1,1)	(E,1,0, 0,2,1)	(1,2,1, E,1,0)	(0,1,1, 1,2,1)	(E,1,0, 1,2,1)	(0,1,1, 0,2,1)	(0,2,1, E,1,0)	(1,2,1, 0,1,1)
Connection Selected	1	2	1	2	2	2	1	1

2. Find the wait time for each state in each group for the connection selected. The wait time is simply the value of the variable a .

3. Further divide the states in each group into subgroups according to their wait time.

4. Add the probabilities of states in each subgroup, and then divide the sum by the corresponding total probability for the group. The resulting value is the probability that the wait time is equal to the wait time of the subgroup.

For example, to find connection 1's wait time distribution,

$$P(\text{wait} = 0 \text{ in queue 1}) = (P_1 + P_7) / (P_1 + P_3 + P_7 + P_8) = 0.45 / 0.50 = 0.90$$

$$P(\text{wait} = 1 \text{ in queue 1}) = (P_3 + P_8) / (P_1 + P_3 + P_7 + P_8) = 0.05 / 0.50 = 0.10$$

The results of both the theoretical approximation and simulation are tabulated in Table 4.4. Note that since the maximum value of a used in the theoretical approximation is one, there is no information for wait time larger than one, thus N.A. is assigned to entries under the Theoretical Approximation column with wait time greater than one. According to the simulation results, there is a small probability that a cell may have wait times of two or three.

The following steps are used to find the cdv-count distribution:

1. Divide all the states into two groups based on the connection selected for transmission.

Table 4.4 Comparison of Results for the Two-Connection System Example

Wait Time	Theoretical Approximation		Simulation	
	Queue 1	Queue 2	Queue 1	Queue 2
0	0.90	0.90	0.93808	0.93454
1	0.10	0.10	0.06101	0.06433
2	N.A.	N.A.	0.00084	0.00106
3	N.A.	N.A.	0.00007	0.00007

N.A.: Not Applicable

2. Find the cdv-count value for the connection in each state in each group. The cdv-count value is simply the difference between the average cell emission period (T) and lt .

3. Further divide the states in each group into subgroups, based on the cdv-count values.

4. Add the probabilities of states in each subgroup, and then divide the sum by the corresponding probability for the group. The resulting value is the probability of cdv-count value for the connection.

For example, with the sample case, the cdv-count distribution for the connection 1 is

$$\text{Probability (cdv-count} = 10 - 1 = 9) = \frac{0}{(0.450 + 0.045 + 0.005)} = 0$$

$$\text{Probability (cdv-count} = 10 - 2 = 8) = \frac{(0.450 + 0.045 + 0.005)}{(0.450 + 0.045 + 0.005)} = 1$$

The results of both the theoretical approximation and simulation are tabulated in Table 4.5. Note that the theoretical approximation indicates, with a probability of 1, that the cell's cdv-count for both connections is equal to 8, while the simulation presents a much different distribution. The reason for this discrepancy is that the original search space is limited to the maximum value of lt of 2 for both connections, while the actual lt is mostly greater than 2 due to the low transmission rates ($\lambda = 0.1$ for both connections.)

Based on the rules developed in this section, a computer program is written to generate a system of equations for the Markov chain approximation. One of the equations in the system of equations is then

Table 4.5 Comparison of Results For The Two-Connection System Example

cdv-count	Theoretical Approximation		Simulation	
	Queue 1	Queue 2	Queue 1	Queue 2
9	0.00	0.00	0.08632	0.08632
8	1.00	1.00	0.09818	0.09818
7	N.A.	N.A.	0.08533	0.08533
≤ 6	N.A.	N.A.	0.72996	0.72996

N.A.: Not Applicable

replaced by Eqn. 4.5, and the resulting system of equations is solved by MATLAB. The results for the values of a ranging from 1 to 2, and that of lt ranging from 2 to 10 are tabulated in Tables 4.6 and 4.7. The simulation results are also listed for comparison.

From Table 4.6 and Table 4.7, it can be observed that as the search space expands, the probability values approach the values obtained from simulation. However, the number of states generated also rapidly increases. In fact, due to the explosion of the number of states, for the two-connection system example, the computer program written can handle only up to search spaces of $[(3,5), (3,5)]$. Memory runs out if search spaces larger than $[(3,5), (3,5)]$ are attempted.

Another observation made is that, for two approximations with different search spaces, the rate of change in the probability values can function as a good indicator of the appropriateness of the size of the search space used. For example, the difference in probabilities of wait time of two between the search space of $[(2, 2), (2, 2)]$ and $[(2, 3), (2, 3)]$ is about 0.007; by the time state $[(2, 10), (2, 10)]$ is calculated, the difference between $[(2, 9), (2, 9)]$ and $[(2, 10), (2, 10)]$ is only 0.0005. Thus, it is conceivable to solve the system by repeatedly applying the Markov Chain approximation with slightly larger search space until the difference in the limiting probabilities calculated is less than a pre-defined value.

4.6 Summary

In this chapter, the formulation of the new algorithm is presented. There are three policies in the new algorithm: the buffer management policy, the discard policy, and the transmission policy. The buffer management policy is adopted from papers presented by Chao, and is not analyzed in this research. The discard policy has two stages: the first stage is used to regulate cell arrival rates and protect compliant

connections, while the second stage tries to discard cells so that the pairwise ratios of the actual cell loss ratios approaches that of the specified ACLR's. The transmission policy utilizes the time-varying priority concept, and selects a cell with the most urgent combined wait-time and cdv-count priority for transmission. A Markov chain approximation method is also presented for determination of theoretical results for the transmission policy.

Table 4.6 Theoretical Approximation Under Different Search Space for Two-Connection System

Search Space	(1, 2)	(1, 3)	(1, 4)	(1, 5)	(1, 6)	(1, 7)	(1, 8)	(1, 9)	(1, 10)	(1, 10)	Simulation
# of States Generated	8	30	60	98	144	198	260	330	408	408	N.A.
P (No Transmission)	0.00000	0.28360	0.43020	0.51920	0.57860	0.62080	0.65160	0.67560	0.69420	0.69420	N.A.
P(wait=0)	0.90000	0.89732	0.90165	0.90482	0.90888	0.91231	0.91585	0.91703	0.91793	0.91793	0.93707
P(wait=1)	0.10000	0.10268	0.09835	0.09518	0.09112	0.08769	0.08415	0.08297	0.08207	0.08207	0.06205
P(wait=2)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.00077
P(wait=3)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.00011
P(cdv-count=0)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.39002	0.39002	0.38173
P(cdv-count=1)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.43282	0.04268	0.04268	0.04173
P(cdv-count=2)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.48300	0.04768	0.04924	0.04924	0.04865
P(cdv-count=3)	N.A.	N.A.	N.A.	N.A.	N.A.	0.53671	0.05187	0.05387	0.05384	0.05384	0.05381
P(cdv-count=4)	N.A.	N.A.	N.A.	N.A.	0.59848	0.05811	0.05937	0.06006	0.05844	0.05844	0.06227
P(cdv-count=5)	N.A.	N.A.	N.A.	0.66833	0.06407	0.06603	0.06628	0.06687	0.06763	0.06763	0.06380
P(cdv-count=6)	N.A.	N.A.	0.74851	0.07107	0.07546	0.07448	0.07493	0.07307	0.07354	0.07354	0.07797
P(cdv-count=7)	N.A.	0.84375	0.07938	0.08354	0.08306	0.08452	0.08300	0.08297	0.08339	0.08339	0.08533
P(cdv-count=8)	1.00000	0.10352	0.10678	0.10474	0.10299	0.10143	0.10086	0.09907	0.09783	0.09783	0.09818
P(cdv-count=9)	N.A.	0.05273	0.06533	0.07232	0.07594	0.07871	0.08069	0.08359	0.08339	0.08339	0.08632

* N.A.: Not Applicable.

Table 4.7 Theoretical Approximation Under Different Search Space for Two-Connection System

Search Space	(2, 2)	(2, 2)	(2, 3)	(2, 3)	(2, 4)	(2, 4)	(2, 5)	(2, 5)	(2, 6)	(2, 6)	(2, 7)	(2, 7)	(2, 8)	(2, 8)	(2, 9)	(2, 9)	(2, 10)	(2, 10)	Simulation
# of States Generated	30	96	166	242	326	418	518	626	742	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
P (No Transmission)	0.00000	0.28280	0.42960	0.51860	0.57760	0.62000	0.65160	0.67540	0.69432	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
P(wait=0)	0.88982	0.89481	0.89993	0.90417	0.90793	0.91130	0.91494	0.91651	0.91775	0.93707									
P(wait=1)	0.09898	0.10212	0.09867	0.09458	0.09065	0.08765	0.08391	0.08225	0.08054	0.06205									
P(wait=2)	0.01120	0.00307	0.00140	0.00125	0.00142	0.00106	0.00115	0.00124	0.00171	0.00077									
P(wait=3)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.00011									
P(cdv-count=0)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.38173									
P(cdv-count=1)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.04173									
P(cdv-count=2)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.04865									
P(cdv-count=3)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.05381									
P(cdv-count=4)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.06227									
P(cdv-count=5)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.06380									
P(cdv-count=6)	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0.07797									
P(cdv-count=7)	N.A.	0.84124	0.07900	0.08333	0.08306	0.08448	0.08276	0.08225	0.08260	0.08533									
P(cdv-count=8)	1.00000	0.10686	0.10955	0.10792	0.10536	0.10348	0.10345	0.10142	0.10181	0.09818									
P(cdv-count=9)	N.A.	0.05190	0.06426	0.07042	0.07451	0.07814	0.08046	0.08287	0.08321	0.08632									

* N.A.: Not Applicable.

CHAPTER 5. SIMULATION MODEL AND RESULTS

Simulation is used to study the behavior of the algorithm (referred to as the “new algorithm”) proposed in Chapter 4. In this chapter, a simulation model and results are presented. Also, the performance of the new algorithm is compared with the first-come-first-served (FCFS) and weighted-round-robin (WRR) policies. This chapter is broken into six sections. Section 1 describes the development of the simulation model and steps implemented to ensure the model’s accuracy. Section 2 studies the effect of various parameters of the transmission policy. Section 3 analyzes the discard policy. Section 4 compares the behavior of the transmission policy of the new algorithm to FCFS and WRR. Section 5 compares the behavior of the discard and buffer management policies and a overflow discard policy with fixed-size buffers. Section 6 gives a summary of the chapter. The behavior of the buffer management policy is not analyzed in this chapter, as such an analysis has been presented and references given by Chao in [43, 55-56].

5.1. The Simulation Model

The assumptions made in Chapter 4 are observed while creating the simulation model. The simulation model is divided into three modules: arrival generation, service execution, and statistics collection. Figure 5.1 illustrates a typical simulation run. The arrival sequence for each connection is created by the arrival generation module. The output from the arrival generation module is then fed into the service policy execution module to determine a cell’s transmission time, or to be discarded. The service execution module may be the new algorithm, FCFS, or WRR. Therefore, the same arrival sequence can be used for the three different policies if a comparison among them is desired. The statistics collection module is then used to calculate the average number of cells discarded, the mean and variance of the number of cells having a given wait time, and the mean and variance of the number of cells having a given cdv-count value. The probability distributions of the wait time and cdv-count can then be calculated from the statistics collected.

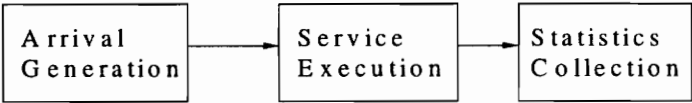


Figure 5.1 Simulation Model

Each simulation run consists of thirty replications. A replication is a series of arrivals and services generated with a single random number seed. The replications within a simulation run differ from each other in only the random number seeds used. For each replication, the run-length is set to be either 5000 cell time units or 300 times the value of the largest peak emission interval, whichever is larger. The choice of the 300 times the value of the largest peak emission interval is related to the issue of simulation warm-up period and is discussed in the next few paragraphs. At the beginning of each replication, the system is assumed to be completely empty, and the previous cell's transmission time is assumed to be at zero.

The simulation runs are treated as non-terminating simulations, because there is no natural run length which can be specified. Associated with the non-terminating simulations are steady-state parameter values. For the case on hand, these parameters are the means and variances of the probabilities that cells have specific wait time and cdv-count values. However, in the non-terminating simulations, the initial system setup can affect the estimations of the steady-state parameters. Thus, the assumptions of an empty system at the beginning of each replication and previous cell's transmission at zero impact the estimates for the mean and variance of the distributions of wait time and cdv-count. To counter the transient effect of the assumptions made regarding the system's initial conditions, data from a warm-up period is ignored and statistics for data after the warm-up period ends is collected. The duration of the warm-up period for each simulation run is determined based on Welch's Approach (see [58] for detailed discussion) and implemented as follows.

In each simulation run, the duration, starting from the beginning of the simulation run and ending at 300 times the largest value of peak emission interval from among all connections, is used to calculate the size of the warm-up period. This duration is further divided into 10 equal partitions; therefore, the length of each partition is equal to 30 times the largest value of peak emission intervals. Theoretically, there should be about 30 arrivals from the connection with the smallest specified transmission rate (the specified transmission rate is equal to the inverse of the peak emission interval), while more arrivals take place for connections with higher specified transmission rates. The logic behind using this partition length is that, to calculate distributions with any statistical significance, a certain number of data points has to be available. The larger the number of data points, the better the estimate is. After experimenting with different simulation runs, it is determined that a partition length of 30 times the peak emission interval of the connection with the slowest transmission rate provides enough data points to calculate means and variances of probability distributions for the connection with the slowest transmission rate. Consequently, the partition size is chosen to be 30 times the largest peak emission interval.

The wait time and cdv-count probability distributions in each partition are then calculated based on 30 replications; also, the moving averages of these distributions are determined by finding the mean of the distributions for the partition of interest and the two closest partitions (before and after the partition of interest). For the first partition, the moving average is set to be the actual distribution of the partition. The end of the warm-up period is chosen as the end of a partition, such that the differences in probabilities of the moving averages for wait time and cdv-count distributions of any two consecutive partitions after that point are within the range of $(-0.01, 0.01)$. The first partition is always used as a part of the warm-up period; thus, if the first partition satisfies the condition of $(-0.01, 0.01)$, the warm-up period begins at the end of the first partition, rather than at time zero.

The verification of the simulation model is performed via the manual-trace of a small sample run. A simulation run of a two-connection system is used. The simulation run consists of only three replications with run lengths of thirty cell time units. During the trace, the arrival sequence created by the arrival generation module is checked to ensure that the transmission rates are within the values specified, the wait time and cdv-count of cells after passing through the service execution module are checked, the conditions which influence the discarding of cells are examined, and finally, the mean and variance calculated by the statistics collection module is confirmed by manual calculations.

Once the simulation model is verified, the model is validated by comparing the theoretical approximation of a two-connection system with the corresponding simulation results. The simulation results for wait time and cdv-count, with respective confidence intervals are presented in Table 5.1a and 5.1b. The theoretical approximation presented in Chapter 4 is also depicted in Table 5.1 for comparison. The theoretical approximation in Table 5.1 corresponds to the search space of $(2, 10, 2, 10)$ (see Chapter 4). For the wait time distribution, only the theoretical approximation for a wait time of zero is within the confidence intervals. The discrepancy can be attributed to the facts that the theoretical approximation only provides estimated values and that the sum of all probabilities is one. Thus, if the theoretical approximation underestimates the probability of wait time equal to zero, the error has to be distributed among the probabilities of the other wait times, since the probabilities have to add up to one. In the two-connection system example, the cells have an overwhelmingly large probability of a wait time of zero. The error in under-estimating the probability of a wait time of zero is small (in percentage term) with respect to the probability of a wait time of zero, but is large when compared to the probabilities of other wait time values, thereby forcing the result of the theoretical approximation for other wait times to be outside the confidence intervals. A clue supporting this theory is that the probabilities for wait times of two and three calculated by the theoretical

Table 5.1 Validation of the Simulation Model

Wait Time	Simulation Mean	Confidence Interval (99% Confidence)	Theoretical Approximation
0	0.93454	(0.91904, 0.95005)	0.91775
1	0.06433	(0.05938, 0.06928)	0.08054
2	0.00106	(0.00044, 0.00167)	0.00171
3	0.00007	(-0.00007, 0.00021)	N.A.

(a) Wait Time

N.A.: not applicable

cdv-count	Simulation Mean	Confidence Interval (99% Confidence)	Theoretical Approximation
<= 0	0.38502	(0.37928, 0.39077)	0.38786
1	0.04114	(0.03764, 0.04463)	0.04242
2	0.04937	(0.04462, 0.05411)	0.04809
3	0.04999	(0.04553, 0.05446)	0.05366
4	0.05843	(0.05410, 0.06276)	0.05973
5	0.06680	(0.06136, 0.07224)	0.06651
6	0.07447	(0.06908, 0.07985)	0.07410
7	0.08479	(0.07964, 0.08993)	0.08260
8	0.10159	(0.09576, 0.10742)	0.10181
9	0.08841	(0.08350, 0.09333)	0.08321

(b) cdv-count

approximation are both larger than the corresponding values from simulation. Another factor that may have affected the results of the theoretical approximation is that the omission of a wait time of three. The theoretical approximation stops at a wait time of two (thus the N.A. in the theoretical column); attempt to perform the theoretical approximation for the wait time of three failed due to the heavy demand of the computer memory, as discussed in Chapter 4.

The probabilities for cdv-count, except for the cdv-count of nine, are completely within the confidence intervals. The possible reasons for the discrepancy are similar to that given earlier: error shifting due to the sum of all probabilities is one and omissions of possible cdv-count values. Note that the results listed in the

theoretical approximation column has zero as the minimum value of cdv-count (or $lt = 10$, see Chapter 4), while the simulation reveals data points with negative cdv-count values.

Note that both the theoretical approximation approach and the simulation provide only an estimate of the real distribution, and the real distribution is not known. Thus, no comparison can be made between results generated by these two approaches and a “correct” solution, and the simulation model can not be truly validated. However, the fact that two completely different approaches provide similar results provides strong proof that the simulation model is correct.

The simulation experiments are designed to examine the behavior of the new algorithm under various conditions. The experiments are divided into the following four parts: part one studies the effects of individual parameter except ACLR, part two analyzes the actual loss ratios of connections under different loads, part three compares the behavior of the new algorithm with FCFS and WRR under different configurations, and part four compares the cell discarding behavior of the new algorithm against the overflow discard policy with fixed-size buffers. The results for these four parts are presented in Section 5.2, Section 5.3, Section 5.4, and Section 5.5, respectively. For all simulation runs, unless specified otherwise, the values used for α , β , c_1 , and c_2 are 1, 1, 2, and 2, respectively. Table 5.2 summarizes the objectives of simulation runs; also listed in Table 5.2 are the figures and tables associated with each simulation run.

5.2. Impact of Individual Parameters in Transmission Policy

To study the impact of individual parameters on the behavior of the new algorithm, seven simulation runs are carried out. The simulation runs are designed to identify the effect that individual parameters have on the system. The first four runs use a three-connection system, while the remaining three runs use a four-connection system. As shown in the Table 5.2, the first four simulation runs are designed to find the impact of parameters MPD, T, and ACDV on the system; the configurations are: 1A, three connections with the same parameter values; 1B, three connections with only the MPD varied; 1C, three connections with only the T varied; and 1D, three connections with only the ACDV varied. The remaining three runs are designed to show the effect of the cost factors, c_1 and c_2 . Their configurations are: 1E, four connections with $c_1 = c_2 = 2.0$; 1F, four connections with only $c_1 = 200$; and 1G, four connections with only $c_2 = 200$. In order to isolate the effect of the individual parameters, the buffer size is set to 200, thereby implying that no cell will be discarded. The actual parameters used in these runs are given in Table 5.3.

Table 5.2 Objectives of Simulation Runs

Simulation Runs	Objective	Results
1A	To study the transmission policy's behavior in a homogeneous environment	Table 5.4, Figure 5.2
1B	To study the impact of varying MPD values on the transmission policy	Table 5.5, Figure 5.3
1C	To study the impact of varying T values on the transmission policy	Table 5.6, Figure 5.4
1D	To study the impact of varying ACDV values on the transmission policy	Table 5.7, Figure 5.5
1E, 1F, 1G	To study the impact of cost factors, c_1 and c_2 , on the transmission policy	Table 5.8, Figure 5.6, Table 5.9, Figure 5.7
2A	To study the discard policy's behavior in a homogeneous environment	Table 5.11
2B	To study the impact of varying ACDV values on the discard policy	Table 5.12
2C	To study the behavior of the discard policy when actual transmission rates are higher than the rates specified in traffic contracts	Table 5.13
2D	To study the impact of varying buffer size on the discard policy with some connections transmitting at higher rates than specified	Table 5.14
3A	To study the comparative performance of the transmission policy of the new algorithm, FCFS, and WRR	Table 5.16, Figure 5.10, Figure 5.11
4A, 4B	To study the comparative performance of the discard and management policies of the new algorithm and overflow discard policy with fixed-size buffers	Table 5.18

The results of simulation runs are presented in Figures 5.2, 5.3, 5.4, and 5.5. In the figures for cdv-count, the zero columns include all cells which have cdv-count values of zero or less. The notation used in the legend has a form of "NEW-#". The designation "NEW" stands for the new algorithm, and the "#" indicates the connection from which data is collected. The numerical data is presented in Tables 5.4, 5.5, 5.6, and 5.7.

Figure 5.2 shows the behavior of the new algorithm in a homogeneous environment. Both the wait time and the cdv-count distributions for all three connections are approximately equal, indicating the new algorithm allocates available resources fairly among connections. Approximately forty percent of cells for all three connections are transmitted with cdv-count of zero or less, while the remaining sixty percent of cells is

Table 5.3 Parameters Used to Study the Effect of Individual Parameter

Run	Connections	MPD	T	ACDV	ACLR
1A	1	5	5	5	0.000001
	2	5	5	5	0.000001
	3	5	5	5	0.000001
1B	1	5	5	5	0.000001
	2	25	5	5	0.000001
	3	125	5	5	0.000001
1C	1	5	5	5	0.000001
	2	5	25	5	0.000001
	3	5	125	5	0.000001
1D	1	5	5	1	0.000001
	2	5	5	5	0.000001
	3	5	5	15	0.000001
1E, 1F, 1G	1	10	5	3	0.000001
	2	5	5	1	0.000001
	3	1	5	3	0.000001
	4	0	5	1	0.000001

Table 5.4 Simulation Results, Run 1A

	New-1	New-2	New-3
P(wait=0)	0.66951	0.66731	0.66493
P(wait=1)	0.20584	0.20617	0.20709
P(wait=2)	0.08694	0.08860	0.08979
P(wait=3)	0.02676	0.02704	0.02658
P(wait=4)	0.00804	0.00837	0.00851
P(wait=5)	0.00216	0.00196	0.00259
P(wait=6)	0.00072	0.00051	0.00048
P(wait=7)	0.00003	0.00003	0.00003

(a) Wait Time Distribution

	New-1	New-2	New-3
P(cdv-count<=0)	0.39123	0.39295	0.39021
P(cdv-count=1)	0.11702	0.11427	0.11478
P(cdv-count=2)	0.18391	0.18651	0.18639
P(cdv-count=3)	0.18942	0.19186	0.19143
P(cdv-count=4)	0.11842	0.11441	0.11719

(b) cdv-count Distribution

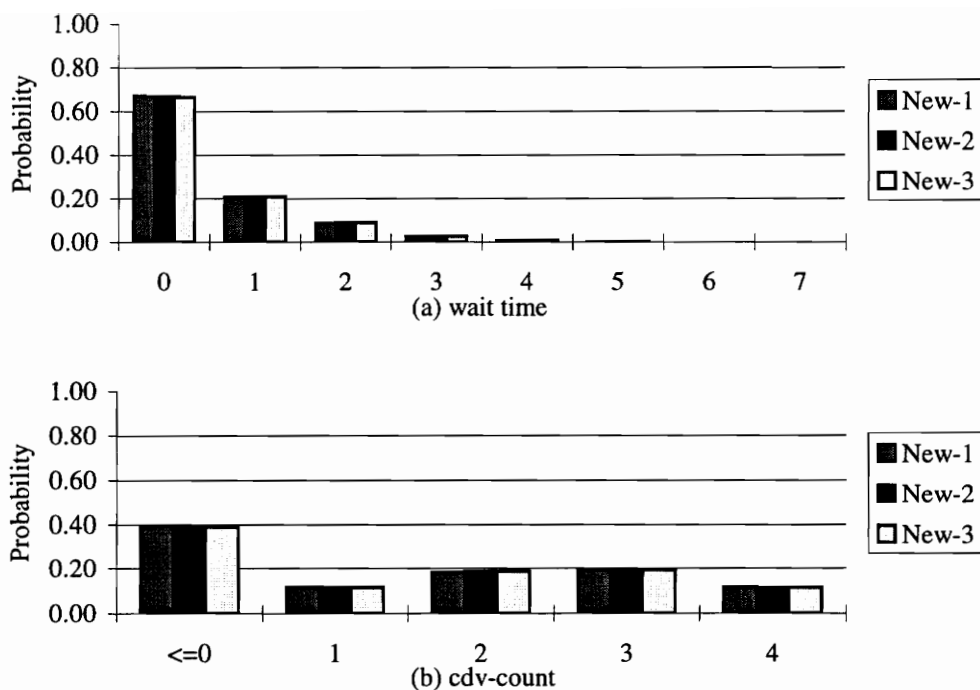


Figure 5.2 Simulation Results, Run 1A

distributed among other possible cdv-count values. The percentage of cells having to wait longer than the MPD is negligible. The actual data is presented in Table 5.4.

Figure 5.3 shows the impact that different values of MPD have on the system. All three connections are given the same characteristic parameters except for the MPD value, where connection 2 has five times as much MPD as connection 1, and connection 3 has five times as much as connection 2. Table 5.5 (a) and Figure 5.3 (a) clearly show that the connection with the smaller MPD value is given a higher priority to transmit than ones with higher MPD values. At the same time, the cdv-count for all three connections are maintained to be approximately equal, which is desirable as the parameter ACDV and T are the same for all three connections. Again, approximately forty percent of cells have cdv-count value of zero or less, while the remaining sixty percent distributed among possible cdv-count values.

The result of Run 1C is depicted in Figure 5.4 and tabulated in Table 5.6. In this run, the parameter that is varied is T, the peak emission interval. The parameter T is varied by a factor of five: connection 1 has a T of 5 cell time units, connection 2 has 25, and connection 3 has 125. Note that varying T has the same effect as changing the peak transmission rate for a connection, since T is the inverse of PCR. Figure 5.4 (a) shows that the connection with a small value of T (i.e., connection 1) is given slight priority over the other

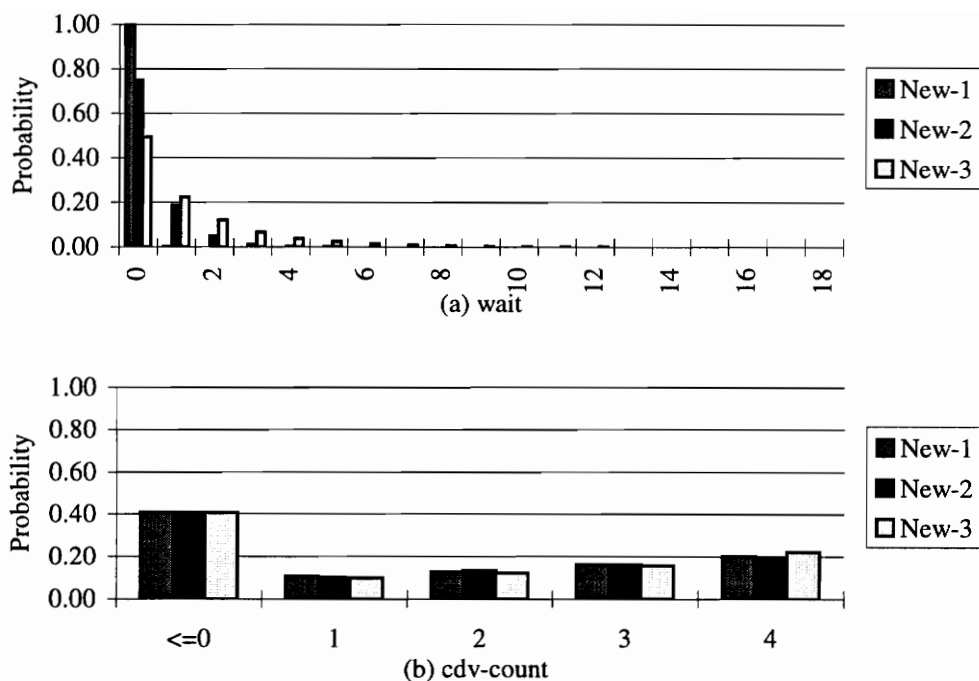


Figure 5.3 Simulation Results, Run 1B

Table 5.5 Simulation Results, Run 1B

	New-1	New-2	New-3		New-1	New-2	New-3
P(wait=0)	0.99808	0.74996	0.49401	P(wait=10)	0.00000	0.00000	0.00184
P(wait=1)	0.00192	0.18549	0.22247	P(wait=11)	0.00000	0.00000	0.00119
P(wait=2)	0.00000	0.04897	0.12075	P(wait=12)	0.00000	0.00000	0.00075
P(wait=3)	0.00000	0.01133	0.06565	P(wait=13)	0.00000	0.00000	0.00041
P(wait=4)	0.00000	0.00329	0.03801	P(wait=14)	0.00000	0.00000	0.00017
P(wait=5)	0.00000	0.00089	0.02488	P(wait=15)	0.00000	0.00000	0.00010
P(wait=6)	0.00000	0.00003	0.01365	P(wait=16)	0.00000	0.00000	0.00007
P(wait=7)	0.00000	0.00003	0.00813	P(wait=17)	0.00000	0.00000	0.00007
P(wait=8)	0.00000	0.00000	0.00524	P(wait=18)	0.00000	0.00000	0.00003
P(wait=9)	0.00000	0.00000	0.00259				

(a) Wait Time Distribution

	New-1	New-2	New-3
P(cdv-count=0)	0.40653	0.40873	0.40549
P(cdv-count=1)	0.10598	0.10045	0.09689
P(cdv-count=2)	0.12607	0.13240	0.12047
P(cdv-count=3)	0.16104	0.16088	0.15573
P(cdv-count=4)	0.20039	0.19754	0.22141

(b) cdv-count Distribution

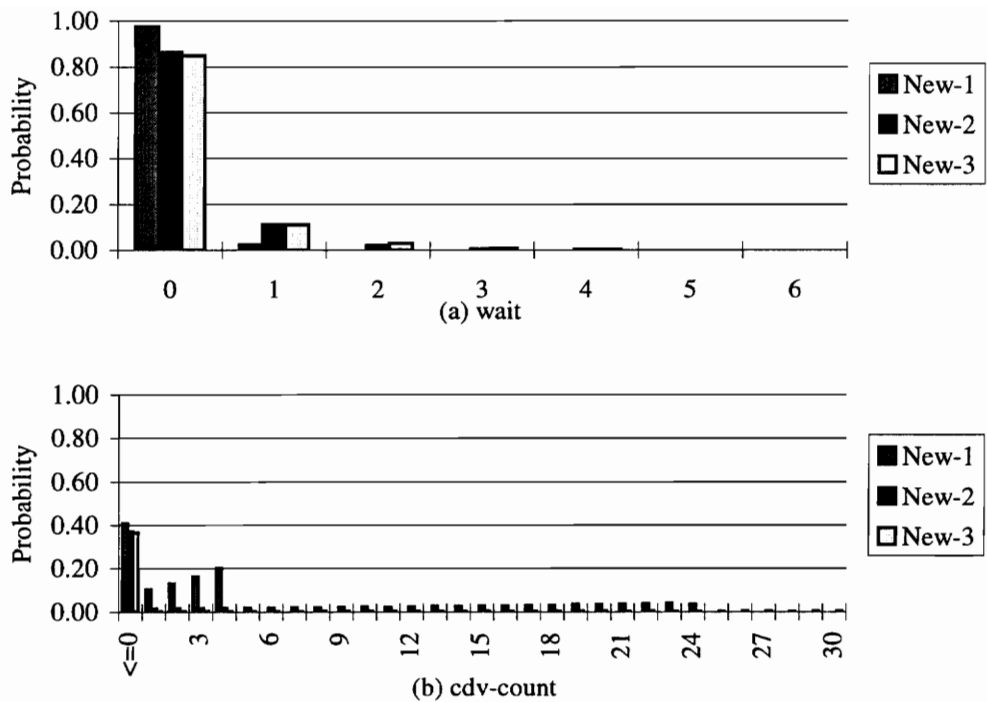


Figure 5.4 Simulation Results, Run 1C

Table 5.6 Simulation Results, Run 1C

	New-1	New-2	New-3
P(wait=0)	0.97642	0.86288	0.84903
P(wait=1)	0.02341	0.11159	0.11021
P(wait=2)	0.00017	0.01983	0.02920
P(wait=3)	0.00000	0.00432	0.00833
P(wait=4)	0.00000	0.00115	0.00224
P(wait=5)	0.00000	0.00017	0.00062
P(wait=6)	0.00000	0.00005	0.00037

(a) Wait Time Distribution

	New-1	New-2	New-3		New-1	New-2	New-3
P(cdv-count=0)	0.40696	0.37180	0.36581	P(cdv-count=6)	0.00000	0.01883	0.00360
P(cdv-count=1)	0.10363	0.01618	0.00311	P(cdv-count=7)	0.00000	0.02050	0.00236
P(cdv-count=2)	0.12857	0.01652	0.00298	P(cdv-count=8)	0.00000	0.02104	0.00447
P(cdv-count=3)	0.16106	0.01765	0.00373	P(cdv-count=9)	0.00000	0.02226	0.00422
P(cdv-count=4)	0.19977	0.01812	0.00323	P(cdv-count=10)	0.00000	0.02325	0.00509
P(cdv-count=5)	0.00000	0.01915	0.00323	P(cdv-count=11)	0.00000	0.02256	0.00398

(b) cdv-count Distribution

continued on next page

	New-1	New-2	New-3		New-1	New-2	New-3
P(cdv-count=12)	0.00000	0.02352	0.00224	P(cdv-count=56)	0.00000	0.00000	0.00497
P(cdv-count=13)	0.00000	0.02592	0.00273	P(cdv-count=57)	0.00000	0.00000	0.00460
P(cdv-count=14)	0.00000	0.02632	0.00286	P(cdv-count=58)	0.00000	0.00000	0.00410
P(cdv-count=15)	0.00000	0.02803	0.00410	P(cdv-count=59)	0.00000	0.00000	0.00621
P(cdv-count=16)	0.00000	0.02821	0.00311	P(cdv-count=60)	0.00000	0.00000	0.00534
P(cdv-count=17)	0.00000	0.02992	0.00385	P(cdv-count=61)	0.00000	0.00000	0.00497
P(cdv-count=18)	0.00000	0.02995	0.00373	P(cdv-count=62)	0.00000	0.00000	0.00547
P(cdv-count=19)	0.00000	0.03478	0.00410	P(cdv-count=63)	0.00000	0.00000	0.00522
P(cdv-count=20)	0.00000	0.03446	0.00485	P(cdv-count=64)	0.00000	0.00000	0.00609
P(cdv-count=21)	0.00000	0.03704	0.00261	P(cdv-count=65)	0.00000	0.00000	0.00435
P(cdv-count=22)	0.00000	0.03859	0.00435	P(cdv-count=66)	0.00000	0.00000	0.00609
P(cdv-count=23)	0.00000	0.04026	0.00323	P(cdv-count=67)	0.00000	0.00000	0.00323
P(cdv-count=24)	0.00000	0.03515	0.00360	P(cdv-count=68)	0.00000	0.00000	0.00472
P(cdv-count=25)	0.00000	0.00000	0.00335	P(cdv-count=69)	0.00000	0.00000	0.00572
P(cdv-count=26)	0.00000	0.00000	0.00373	P(cdv-count=70)	0.00000	0.00000	0.00659
P(cdv-count=27)	0.00000	0.00000	0.00398	P(cdv-count=71)	0.00000	0.00000	0.00497
P(cdv-count=28)	0.00000	0.00000	0.00335	P(cdv-count=72)	0.00000	0.00000	0.00472
P(cdv-count=29)	0.00000	0.00000	0.00385	P(cdv-count=73)	0.00000	0.00000	0.00435
P(cdv-count=30)	0.00000	0.00000	0.00460	P(cdv-count=74)	0.00000	0.00000	0.00485
P(cdv-count=31)	0.00000	0.00000	0.00447	P(cdv-count=75)	0.00000	0.00000	0.00398
P(cdv-count=32)	0.00000	0.00000	0.00422	P(cdv-count=76)	0.00000	0.00000	0.00572
P(cdv-count=33)	0.00000	0.00000	0.00311	P(cdv-count=77)	0.00000	0.00000	0.00733
P(cdv-count=34)	0.00000	0.00000	0.00447	P(cdv-count=78)	0.00000	0.00000	0.00646
P(cdv-count=35)	0.00000	0.00000	0.00298	P(cdv-count=79)	0.00000	0.00000	0.00621
P(cdv-count=36)	0.00000	0.00000	0.00323	P(cdv-count=80)	0.00000	0.00000	0.00335
P(cdv-count=37)	0.00000	0.00000	0.00348	P(cdv-count=81)	0.00000	0.00000	0.00522
P(cdv-count=38)	0.00000	0.00000	0.00410	P(cdv-count=82)	0.00000	0.00000	0.00746
P(cdv-count=39)	0.00000	0.00000	0.00435	P(cdv-count=83)	0.00000	0.00000	0.00584
P(cdv-count=40)	0.00000	0.00000	0.00335	P(cdv-count=84)	0.00000	0.00000	0.00472
P(cdv-count=41)	0.00000	0.00000	0.00398	P(cdv-count=85)	0.00000	0.00000	0.00398
P(cdv-count=42)	0.00000	0.00000	0.00460	P(cdv-count=86)	0.00000	0.00000	0.00472
P(cdv-count=43)	0.00000	0.00000	0.00485	P(cdv-count=87)	0.00000	0.00000	0.00534
P(cdv-count=44)	0.00000	0.00000	0.00460	P(cdv-count=88)	0.00000	0.00000	0.00621
P(cdv-count=45)	0.00000	0.00000	0.00398	P(cdv-count=89)	0.00000	0.00000	0.00683
P(cdv-count=46)	0.00000	0.00000	0.00410	P(cdv-count=90)	0.00000	0.00000	0.00572
P(cdv-count=47)	0.00000	0.00000	0.00460	P(cdv-count=91)	0.00000	0.00000	0.00534
P(cdv-count=48)	0.00000	0.00000	0.00398	P(cdv-count=92)	0.00000	0.00000	0.00708
P(cdv-count=49)	0.00000	0.00000	0.00460	P(cdv-count=93)	0.00000	0.00000	0.00596
P(cdv-count=50)	0.00000	0.00000	0.00522	P(cdv-count=94)	0.00000	0.00000	0.00621
P(cdv-count=51)	0.00000	0.00000	0.00547	P(cdv-count=95)	0.00000	0.00000	0.00509
P(cdv-count=52)	0.00000	0.00000	0.00348	P(cdv-count=96)	0.00000	0.00000	0.00746
P(cdv-count=53)	0.00000	0.00000	0.00360	P(cdv-count=97)	0.00000	0.00000	0.00646
P(cdv-count=54)	0.00000	0.00000	0.00385	P(cdv-count=98)	0.00000	0.00000	0.00634
P(cdv-count=55)	0.00000	0.00000	0.00572	P(cdv-count=99)	0.00000	0.00000	0.00559

(b) cdv-count Distribution
continued from previous page, continued on next page

	New-1	New-2	New-3		New-1	New-2	New-3
P(cdv-count=100)	0.00000	0.00000	0.00659	P(cdv-count=113)	0.00000	0.00000	0.00746
P(cdv-count=101)	0.00000	0.00000	0.00659	P(cdv-count=114)	0.00000	0.00000	0.00621
P(cdv-count=102)	0.00000	0.00000	0.00733	P(cdv-count=115)	0.00000	0.00000	0.00609
P(cdv-count=103)	0.00000	0.00000	0.00783	P(cdv-count=116)	0.00000	0.00000	0.00944
P(cdv-count=104)	0.00000	0.00000	0.00733	P(cdv-count=117)	0.00000	0.00000	0.00820
P(cdv-count=105)	0.00000	0.00000	0.00584	P(cdv-count=118)	0.00000	0.00000	0.00770
P(cdv-count=106)	0.00000	0.00000	0.00584	P(cdv-count=119)	0.00000	0.00000	0.00708
P(cdv-count=107)	0.00000	0.00000	0.00808	P(cdv-count=120)	0.00000	0.00000	0.00659
P(cdv-count=108)	0.00000	0.00000	0.00795	P(cdv-count=121)	0.00000	0.00000	0.00795
P(cdv-count=109)	0.00000	0.00000	0.00696	P(cdv-count=122)	0.00000	0.00000	0.00845
P(cdv-count=110)	0.00000	0.00000	0.00683	P(cdv-count=123)	0.00000	0.00000	0.00895
P(cdv-count=111)	0.00000	0.00000	0.00746	P(cdv-count=124)	0.00000	0.00000	0.00522
P(cdv-count=112)	0.00000	0.00000	0.00646				

(b) cdv-count Distribution
continued from previous page

connections when the wait time is zero, forcing cells from the other connections to wait slightly longer. Figure 5.4 (b) indicates that approximately forty percent of cells has a cdv-count of zero or less, despite different peak emission intervals; the remaining sixty percent of the cells scatters approximately evenly among the possible cdv-count values for all connections. Note that Figure 5.4 (b) only plots data for the first thirty cdv-count; Table 5.6 (b) shows the complete set of data.

Figure 5.5 and Table 5.7 show the distributions of wait time and cdv-count for Run 1D. In this run, the ACDV parameter is varied. A small ACDV value indicates that the connection's tolerance to the cell delay variation is small, and it is desirable to have cells transmitted near their theoretical transmission times (TTT's). From Figure 5.5 (a), it is seen that connection 1, which has the smallest ACDV, is forced to wait, while connections 2 and 3 have a slightly higher priority in transmission. The wait time distributions of connections 2 and 3 are identical, even though connection 3 has an ACDV of 15, while connection 2 has an ACDV of 5. The similarity in connections 2's and 3's wait time distributions is due to the fact that the values of the parameter T for both connections are 5, which implies that the maximum ACDV-window is 5 cell time units (ctu). Connection 2 has an ACDV of 5, thus its ACDV-window is exactly 5 ctu. Connection 3, on the other hand, has an ACDV of 15, which exceeds the maximum ACDV-window allowed, thereby limiting the actual ACDV-window to 5 ctu. Thus, connection 2 and connection 3 essentially have the same connection parameters. The distribution of cdv-count for all three connections are almost identical, indicating the fairness in allocating cell delay variations. Again, about forty percent of cells has a cdv-count of less than or equal to zero, while the remaining sixty percent is scattered among other possible cdv-count values.

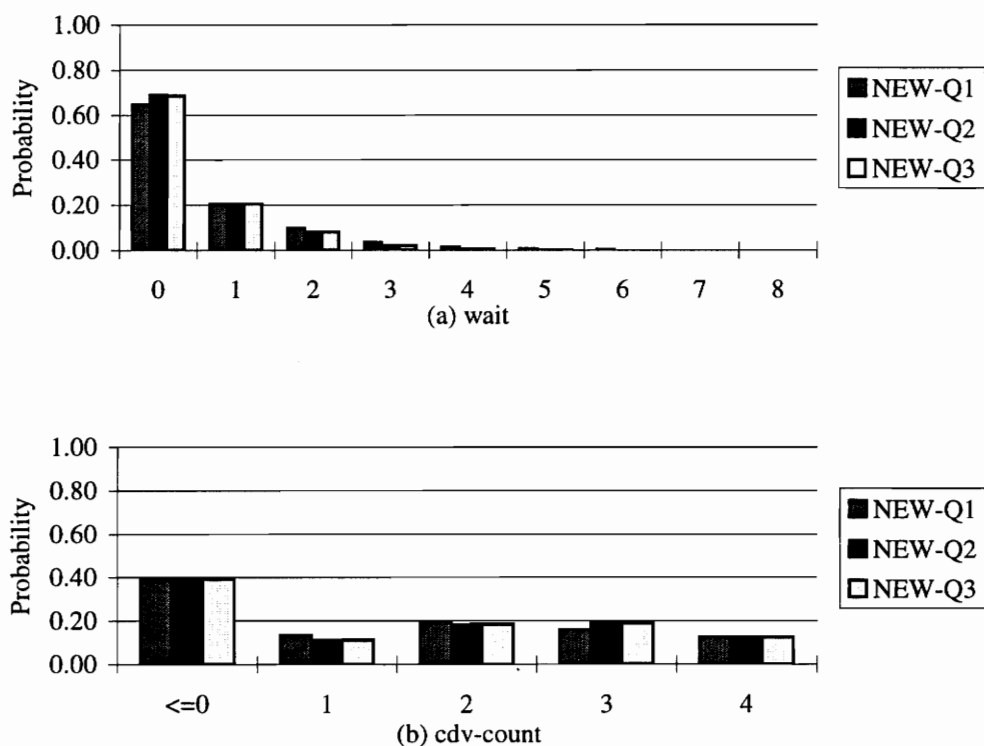


Figure 5.5 Simulation Results, Run 1D

Table 5.7 Simulation Results, Run 1D

	NEW-Q1	NEW-Q2	NEW-Q3
P(wait=0)	0.64514	0.68615	0.68348
P(wait=1)	0.20037	0.20384	0.20487
P(wait=2)	0.09700	0.08150	0.08135
P(wait=3)	0.03391	0.02086	0.02104
P(wait=4)	0.01495	0.00604	0.00688
P(wait=5)	0.00578	0.00117	0.00208
P(wait=6)	0.00219	0.00041	0.00031
P(wait=7)	0.00051	0.00003	0.00000
P(wait=8)	0.00014	0.00000	0.00000

(a) Wait Time Distribution

	NEW-Q1	NEW-Q2	NEW-Q3
P(cdv-count=0)	0.38630	0.39456	0.39110
P(cdv-count=1)	0.13334	0.11101	0.11239
P(cdv-count=2)	0.19534	0.18294	0.18340
P(cdv-count=3)	0.15952	0.18980	0.18857
P(cdv-count=4)	0.12550	0.12168	0.12454

(b) cdv-count Distribution

The results of simulation runs 1E ($c_1 = c_2 = 2$), 1F ($c_1 = 200, c_2 = 2$), and 1G ($c_1 = 2, c_2 = 200$) are consolidated to better present the behavior due to the cost factors. The wait time distributions are plotted in Figure 5.6, while the cdv-count distributions are depicted in Figure 5.7. There are four plots in each figure with each plot presenting the probability distribution of a connection in three different simulation runs. The notation used in the legend has the form of "Run1X-#", where X designates the simulation run, and # is the connection number. The four connections have exactly the same connection characteristic parameters, except for the MPD's. Connection 1's MPD is much larger than its expected peak emission interval (T), connection 2's MPD is equal to its T value, and the MPD's of connections 3 and 4 are less than their respective T values.

Comparing Run 1E and Run 1F in Figure 5.6, it is observed that increasing c_1 from 2 to 200 has little effect on connection 1, as they have nearly identical distributions. Connection 2 shows a small increase of probability at wait time equal to six. Connection 3 shows a slight shift in the distribution toward the wait time of 2 from longer wait times. Connection 4 shows marked improvement as the probability of the wait time of 1 for Run 1F is increased by about 0.1 in comparison to that of Run 1E, while reducing probabilities for longer wait times. The values used for MPD's can shed some light on the behavior of the four connections. Increasing c_1 is essentially equivalent to trying to reduce the number of cells having wait times greater than the specified MPD's. The c_1 does not have any impact until a cell's wait time becomes larger than the specified MPD, which explains the shifting of probabilities at wait times of 1, 2, and 6, for connections 2, 3, and 4, respectively. For connection 1, the cost factor does not become effective until the wait time becomes eleven. However, since the probability of a wait time equal to or larger than eleven in Run 1E is less than 0.050 (see Table 5.8), increasing c_1 has little effect on the overall distribution. Nevertheless, there is a very small shift in probability occurring at a wait time of eleven for Run 1F, as shown in Table 5.8. For the wait time distributions of connections 2, 3 and 4 in Run 1E, on the other hand, the probabilities that cells have wait times larger than the specified MPD's are 0.086, 0.167, and 0.365, respectively. Therefore, the improvement made by increasing c_1 becomes more significant.

Increasing c_2 from 2 to 200, while holding c_1 at 2, has a much different effect on the wait time distributions. In Figure 5.6, connection 1 shows a shift in distribution toward the wait time of zero, connection 2 and 3 show no change, while connection 4 shows a shift toward longer wait times. This shift in distributions is caused by the relative magnitude of the parameters used. In Run 1E, where c_1 and c_2 are equal, cells in connection 1 are forced to wait longer because of the connection 1's larger MPD value, while other connections have higher priorities to transmit. A consequence is that connections with stricter delay

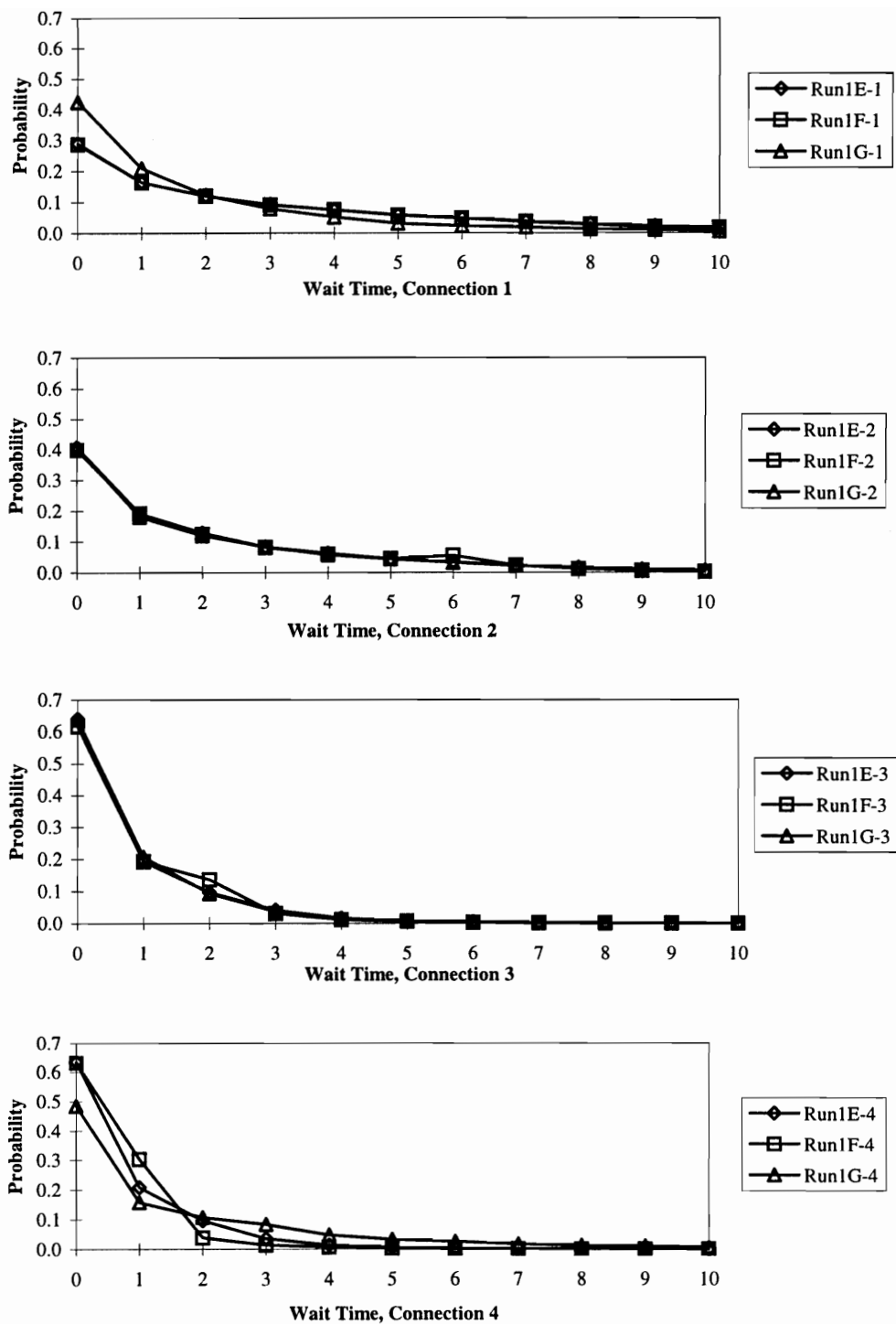


Figure 5.6 Simulation Results, Run 1E, 1F, and 1G, Wait Time Distribution

Table 5.8 Simulation Results, Run 1E, 1F, and 1G, Wait Time, Connection 1

wait time	Run1E-1	Run1F-1	Run1G-1	wait time	Run1E-1	Run1F-1	Run1G-1
0	0.289851	0.287191	0.42499	14	0.005118	0.004774	0.001374
1	0.163902	0.163466	0.208957	15	0.002610	0.002817	0.000859
2	0.119835	0.119603	0.122407	16	0.001477	0.001099	0.000240
3	0.091018	0.090715	0.078445	17	0.000962	0.000962	0.000240
4	0.074532	0.073747	0.051278	18	0.000481	0.000756	0.000034
5	0.057427	0.056951	0.030533	19	0.000172	0.000206	0.000000
6	0.047948	0.046852	0.022874	20	0.000137	0.000069	0.000000
7	0.037609	0.035757	0.017825	21	0.000069	0.000137	0.000000
8	0.029057	0.027204	0.012364	22	0.000636	0.000803	0.000000
9	0.022154	0.019545	0.009548	23	0.000569	0.000401	0.000000
10	0.018204	0.01714	0.006045	24	0.000201	0.000468	0.000000
11	0.016040	0.023941	0.005255	25	0.000100	0.000134	0.000000
12	0.013326	0.017312	0.003984	26	0.000034	0.000034	0.000000
13	0.008071	0.009755	0.002748				

requirements tend to have large portions of cells transmitted outside of the ACDV-window. Thus, when c_2 is increased drastically in Run 1G, the connections with more cells transmitted outside of the ACDV-window are refrained from transmission when other connections have cells with smaller cdv-count waiting in the buffer. As it turns out, the only connections with wait time distributions drastically changed in Run 1G are connections 1 and 4, while connection 2 and 3 remain almost the same. Cells in connection 4, which have the strictest MPD and ACDV requirements, have higher probability of waiting longer; while cells in connection 1, which has the least demanding MPD and ACDV requirements, have higher probability of waiting less.

The increase in c_1 does not affect the cdv-count distribution as much as the increase in c_2 , as shown in Figure 5.7. All four connections in the Run 1E has approximately the same cdv-count distributions as their counterparts in the Run 1F. The connections under Run 1G, however, show drastically different distributions. Because of the large c_2 , all four connections from Run 1G show a shift in probability toward the ACDV's specified: connection 1 and 3 have the probabilities shifted toward a cdv-count of 3, while connection 2 and 4 have the probabilities shifted toward a cdv-count of 1. The most important behavior is that for all four connections, the cumulative probability of the cdv-count being greater than the ACDV value specified is reduced in Run 1G, as shown under the improvement column in Table 5.9.

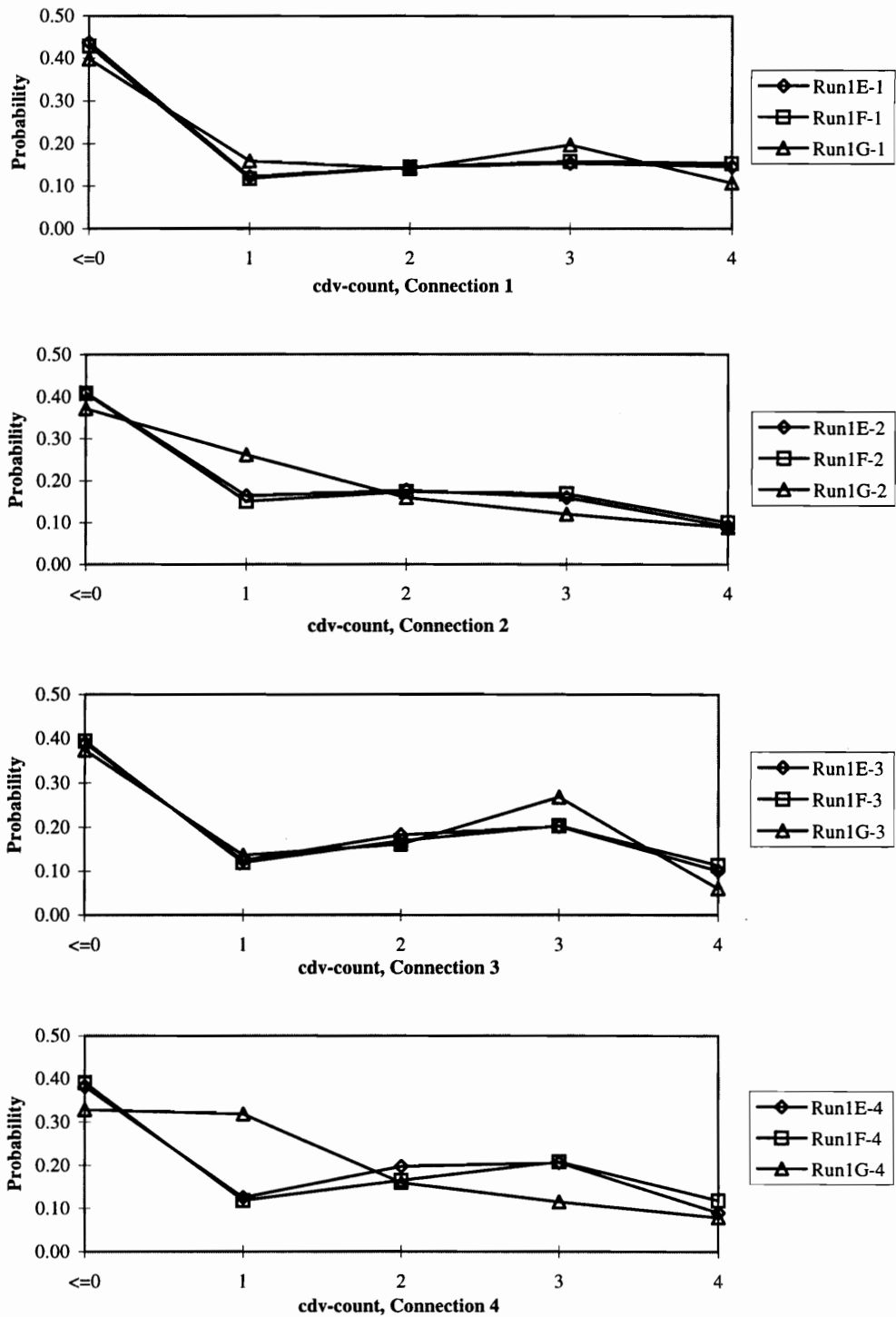


Figure 5.7 Simulation Results, Run 1E, 1F, and 1G, cdv-count Distribution

Table 5.9 Improvement in the Probabilities of Violating the ACDV Specifications

	Run 1E	Run 1G	Improvement
Connection 1	0.14542	0.10678	0.03864
Connection 2	0.42686	0.36783	0.05903
Connection 3	0.10111	0.06035	0.04076
Connection 4	0.49320	0.35323	0.13996

The effect of the connection parameters can now be summarized.

1. In a homogeneous environment, the wait time and the cdv-count are evenly distributed among connections, thus ensuring the fair usage of system resources.
2. If MPD is the only variable, the new algorithm protects the connections with small MPD values by giving them a higher priority for transmission than the connections with large MPD values. This is desirable because cells with a small MPD can afford to spend less time in the system than cells with a large MPD. The cdv-count distribution is roughly distributed evenly among the connections.
3. If T is the only variable, the new algorithm favors connections with smaller values of T by giving them slightly higher priority than connections with large values of T. The distribution of cdv-count is scattered evenly among the allowable values.
4. If ACDV is the only variable, the cells from the connections with smaller ACDV values are held back to decrease the cdv-count, thereby increasing the cells' wait time. If the ACDV value is larger than the T, then it is treated as if ACDV is equals to T. The cdv-count distributions of different connections are about the same.
5. The impact of varying c_1 and c_2 depends highly on the system configuration. A larger c_1 increases the probability of a wait time of (MPD+1) at the connection, while reducing probabilities at wait times greater than (MPD+1). A larger c_2 reduces the probabilities of cdv-count being greater than the specified ACDV; however, it may also cause the connections' probability distributions for wait time to change significantly. .

5.3. The Behavior of The Discard Policy

To study the cell loss ratio behavior due to the algorithm under different conditions, four simulation runs are used. Run 2A is used as the control case, where the basic behavior of the discard policy is analyzed. Run 2B demonstrates the effect that larger ACDV values have on the number of cells discarded due to MGCRA. Run 2C shows the behavior of the discard policy if some connections transmit cells to the system at rates that exceed the rates specified in the traffic contract. Finally, Run 2D demonstrates the effect of buffer size on the number of cells discarded. The parameters used in the simulation runs are listed in Table 5.10. A three-connection system is used. The buffer size is arbitrarily fixed at two cells for Run 2A, 2B, and 2C, which is very small and thus many cells will be discarded. Six different buffer sizes are used in Run 2D, they are 1, 4, 6, 8, 10, and 12. The parameter λ is the ratio of the actual transmission rate over the transmitter's transmission rate. Thus, under the assumption that the transmitter can transmit only a cell in a time slot and compliance with traffic contract, a connection with $T = 4$ cell time units has a λ of 0.25 cells per cell time unit.

Table 5.10 Parameters Used in Simulation Runs for Discard Policy

Run	Buffer Size	Connection	λ	MPD	T	ACDV	ACLR
2A	2	1	0.250	10	4	5	0.000001
		2	0.250	10	4	5	0.000010
		3	0.250	10	4	5	0.000100
2B	2	1	0.250	10	4	10	0.000001
		2	0.250	10	4	10	0.000010
		3	0.250	10	4	10	0.000100
2C	2	1	0.250	10	4	5	0.000001
		2	0.250	10	5	5	0.000010
		3	0.250	10	8	5	0.000100
2D	1, 4,	1	0.250	10	4	5	0.000001
	6, 8,	2	0.250	10	5	5	0.000010
	10, 12	3	0.250	10	8	5	0.000100

The result of Run 2A is presented in Table 5.11. In terms of the total number of cells discarded, connection 1 lost fewer cells than connection 2, and connection 2 lost fewer cells than connection 3; this outcome is expected because of the relative magnitude of the specified ACLR's. A better representation of the discard policy's behavior emerges if the total number of cells discarded is further divided into two groups, discard due to MGCRA and discard due to rotation. From Table 5.11, the number of cells dropped due to MGCRA is approximately equal for all three connections, as it should be since the connections have identical

parameter values. The number of cells dropped due to rotation, on the other hand, forms ratios similar to that of the specified ACLR's. The specified ACLR's imply that for every cell discarded by connection 1, connection 2 should lose 10 cells, and connection 3 should lose 100 cells. The data presented in the Table 5.11 supports this relationship.

The data in Table 5.11 indicates that a large portion of cells is dropped because of the MGCRA, this is caused by the assumption of the Bernoulli arrival process. With the Bernoulli arrival process, the probability that a cell arrives in a connection in a time slot is the same for all time slots, which implies that some cells would arrive much sooner than the TAT. The number of cells lost due to MGCRA should be reduced if the ACDV requirements are relaxed. Table 5.12 presents the result of Run 2B, where the ACDV requirements for the connections in Run 2A are relaxed from 5 to 10. Clearly, relaxing the ACDV requirements does reduce the cell loss due to MGCRA. Again, the number of cells discarded due to MGCRA are roughly equal for all three connections, and the ratio of the number of cells dropped due to rotation from connections 1 and 2 are proportional to that of the specified ACLR's. The number of cells from connection 3 discarded by rotation is not proportional to that of other connections as specified by the ratios of ACLR's. This is caused by inadequate availability of cells from connection 3 when congestion occurs. In other words, after the MGCRA stage, there is not enough cells from connection 3 for the rotation stage to discard a sufficient number of cells to approach the proportions specified by the ratios of ACLR's. Instead, the system reaches an equilibrium in which all cells from connection 3 are discarded by rotation during congestion. In Run 2B, the equilibrium for the number of cells discarded by rotation occurs at the proportions of 1:11:60.

Table 5.11 Simulation Results, Run 2A

	Connection 1	Connection 2	Connection 3
Total Transmitted	1105.333	1101.733	1090.700
Total Discarded	116.967	119.233	136.333
Discard by MGCRA	116.767	117.133	117.100
Discard by Rotation	0.200	2.100	19.233

Table 5.12 Simulation Results, Run 2B

	Connection 1	Connection 2	Connection 3
Total Transmitted	1131.600	1119.933	1085.267
Total Discarded	91.667	102.700	145.000
Discard by MGCRA	90.800	92.933	92.133
Discard by Rotation	0.867	9.767	52.867

It is conceivable that, because of malicious intention or hardware breakdown, a connection may transmit at a much faster rate than the one specified in the traffic contract. The discard policy should be able to protect the connections which are compliant with the contract by discarding more cells from the non-compliant connections. Run 2C tries to simulate the scenario in which some connections are transmitting faster than they should. The peak emission intervals (T) for connections 2 and 3 are declared to be 5 and 8 cell time units, therefore the PCR's (expressed as λ 's, the load levels on the transmitter) should be 0.2 and 0.125 cells per cell time unit. The actual transmission rates are 0.25 and 0.25. The results of Run 2C is tabulated in Table 5.13. Comparing the results of Table 5.13 to that in Table 5.11, the number of cells discarded by the MGCRA in Run 2C is much higher for connections 2 and 3 than those in Run 2A, while connection 1 maintains a similar performance.

Table 5.13 Simulation Results, Run 2C

	Connection 1	Connection 2	Connection 3
Total Transmitted	1111.600	1051.433	1055.400
Total Discarded	112.300	172.433	175.067
Discard by MGCRA	112.300	172.433	175.067
Discard by Rotation	0.000	0.000	0.000

Run 2C depicts the scenario in which some connections transmit at faster rates than specified. If the system resources are strained, the MGCRA steps in and rejects cells which exceed the allowed transmission rates. However, if there are enough system resources, i.e. the buffer size is large, instead of discarding the non-compliant cells, the system resources should be utilized by transmitting the cells. The effect of buffer size on the number of cells discarded is demonstrated in Run 2D, where six different buffers of sizes between 1 to 12 are used. The result is listed in Table 5.14 and illustrated in Figure 5.8 and Figure 5.9. Figure 5.8 clearly shows that the protection required by the compliant connections diminishes with an increase in buffer size. The number of cells discarded by rotation approaches zero at a much faster rate than that by MGCRA, as shown in Figure 5.9, since MGCRA is applied first when buffer becomes full.

An interesting observation can be made when the buffer size is one. The number of cells of connection 3 discarded by rotation is much smaller than those of connection 2, which is opposite of the fundamental logic on which the rotation index method is based. The aberration occurs because cells from connection 3 are arriving at the node at twice of its specified speed, causing roughly 43 percent ($\approx \frac{548.867}{548.867 + 706.867}$) of

Table 5.14 Simulation Results, Run 2D

	Buffer Size	Connection 1	Connection 2	Connection 3	Buffer Size	Connection 1	Connection 2	Connection 3
Total Transmitted	1	1031.600	832.200	689.833	8	1221.567	1220.067	1225.767
Total Discarded		190.733	388.667	537.100		0.867	0.967	1.167
Discard by MGCRA		183.900	318.867	536.400		0.867	0.967	1.167
Discard by Rotation		6.833	69.800	0.700		0.000	0.000	0.000
Total Transmitted	4	1197.5	1195.033	1199.6	10	1222.133	1220.767	1226.567
Total Discarded		24.933	25.933	27.267		0.300	0.267	0.367
Discard by MGCRA		24.933	25.933	27.267		0.300	0.267	0.367
Discard by Rotation		0.000	0.000	0.000		0.000	0.000	0.000
Total Transmitted	6	1217.133	1215.967	1221.667	12	1222.400	1220.967	1226.867
Total Discarded		5.300	5.067	5.267		0.033	0.067	0.067
Discard by MGCRA		5.300	5.067	5.267		0.033	0.067	0.067
Discard by Rotation		0.000	0.000	0.000		0.000	0.000	0.000

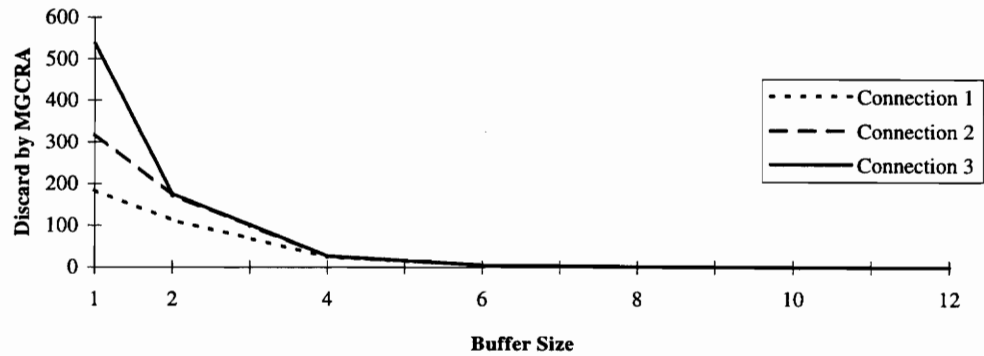


Figure 5.8 Cell Discard Due To MGCRA

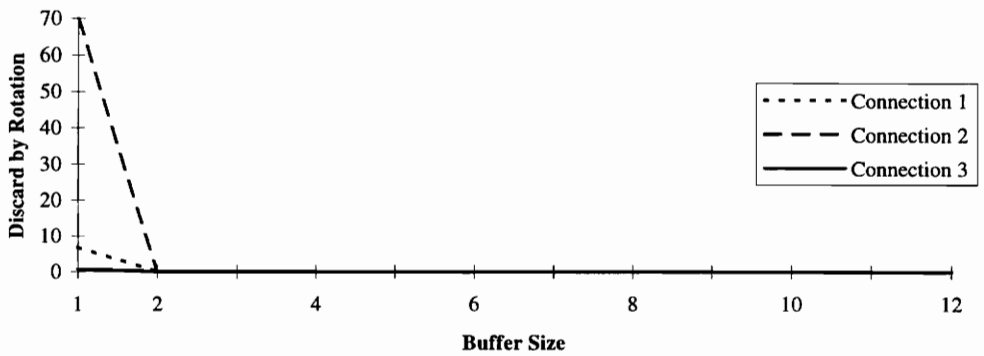


Figure 5.9 Cell Discard Due To Rotation

the cells to be discarded by MGCRA. The cells rarely pass the MGCRA test, thus making it nearly impossible for them to reach the rotation stage of the discard policy, thereby resulting in a low number of cell discard by rotation.

The behavior of the two-stage approach of the discard policy can now be summarized. If all connections are compliant with the specified characteristic parameters, the loss ratio mimics the relative magnitude of the specified ACLR's. The discard policy protects performance of compliant connections if some connections transmit faster than they should. If the system is not congested, transmissions from all connections, which may or may not be compliant with the traffic contract, are allowed for better utilization of the system resources.

5.4. Performance Comparison of Transmission Policies

An important goal of the new algorithm is to find a compromise between the delay and the cell delay variation of the transmitted cells at the node level within an ATM network. In this section, the performance of three transmission policies (the new algorithm, the FCFS, and the WRR) are compared. To make a fair comparison, the discard and the buffer management policies used by the new algorithm are implemented with the FCFS and WRR. Three simulation runs of a six-connection system are carried out, with a different transmission policy used in each run. The parameters used in the system are presented in Table 5.15. The other parameters used include: $c_1 = 2$, $c_2 = 200$, and buffer size of 5.

Table 5.15 Parameters Used To Study Various Transmission Policies

Connection	λ	MPD	T	ACDV	ACLR	Type
1	0.00641	2	156	5	1E-06	Audio
2	0.20000	2	5	2	1E-06	Video
3	0.25000	100	4	100	1E-12	Data
4	0.00641	2	156	5	1E-06	Audio
5	0.20000	2	5	2	1E-06	Video
6	0.25000	100	4	100	1E-12	Data

The six connections are designed to mimic the parameter characteristic of real connections. These six connections can be further divided into three pairs of connections, with each pair representing a type of application. Three possible types of applications are considered, audio, video, and bulk data transfer. The values used in the simulation model are obtained with the following assumptions:

1. The transmitter transmits at 155.52 Mbps, the data rate for OC-3,
2. Audio connections require 1 Mbps with small ACDV's, small MPD's, and a large loss tolerance,
3. Video connections require 31.1 Mbps with small ACDV's, small MPD's, and a large loss tolerance, and
4. Data connections require 38.9 Mbps with large ACDV's, large MPD's, and a small loss tolerance.

The ACDV values are arbitrarily set to 5 for audio connections, 2 for video connections, and 100 for data connections. The MPD values are arbitrarily set to 2 for both audio and video connections, and 100 for data connection. The ACLR values are arbitrarily set to 10^{-6} for both audio and video connections, and 10^{-12} for data connections. The actual transmission rate of each connection is normalized with respect to the transmitter's transmission rate, and is represented by λ . Thus, the transmitter has a resource value of 1, and each connection requires a fraction of the resource as indicated by its value of λ .

The FCFS policy is implemented in the following manner.

The buffer is scanned at the beginning of a time slot. If there is at least one cell present, the cell with the earliest arrival time is selected for transmission. If there are multiple cells with the same arrival time, then the cell with the smallest ACLR is selected. If multiple cells have the same values for ACLR's, then a cell is selected at random.

The WRR policy is implemented in the following manner.

At the beginning of a simulation run, an index for the number of cells to be transmitted from each connection is determined based on the transmission rates. The connection with the lowest transmission rate is given a weight of 1, while other connections are given weights which are the integer portions of the ratios of the connections' transmission rates and the lowest transmission rate. This weighting determines the number of times a connection is scanned for any presence of cells in a WRR cycle. The steps used to generate the sequence from which connections are scanned are

1. Define a connection's increment as the inverse of the connection's weight.
2. Each connection has a index that is used to determine the order of sequence. At the beginning of the simulation run, the index for each connection is set at one half of the connection's increment.
3. At the beginning of the simulation run, the sequence list is empty.
4. Comparing the indices of all connections, the connection with the smallest index is added to the end of the sequence list. If multiple connections have the same index values, the connection with the smallest ACLR is selected first. If multiple connections have the same index values and the same ACLR values, the connection with the smaller channel number is selected. The channel number is the identification number given to each connection in a simulation run.
5. Update the connection's index by adding the connection's increment to the connection's index.
6. Repeat steps 4 and 5 until the number of entries in the sequence list is the same as the length of a WRR cycle.

During the simulation, the presence of cells is checked at the beginning of each time slot. If at least one cell is present, the connections are scanned in the order defined in the sequence list, and the cell from the first non-empty connection is transmitted. When the end of the sequence list is reached, a new sequence starts from the beginning of the sequence list.

The results of the simulation runs for the wait time distributions, the cdv-count distributions and the cell discard behavior are shown in Figure 5.10, Figure 5.11, and Table 5.16, respectively. Note that the figures do not show the entire set of data for all connections. Since two connections of the same type are used in the simulation run and these two connections have the same performance, the average performance of the connection pair is presented instead of giving similar data twice.

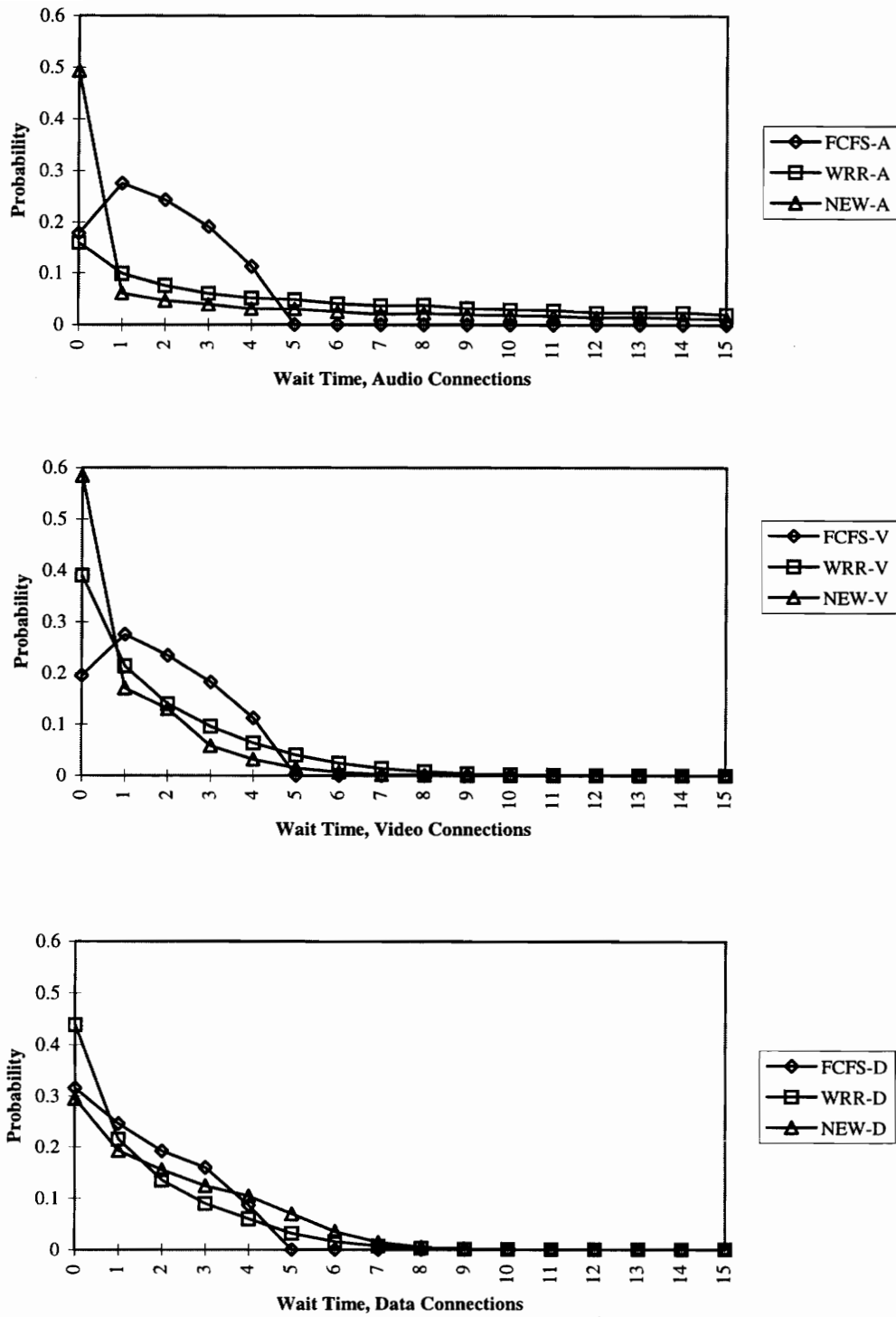


Figure 5.10 Simulation Results, Run 3A, Wait Time Distributions

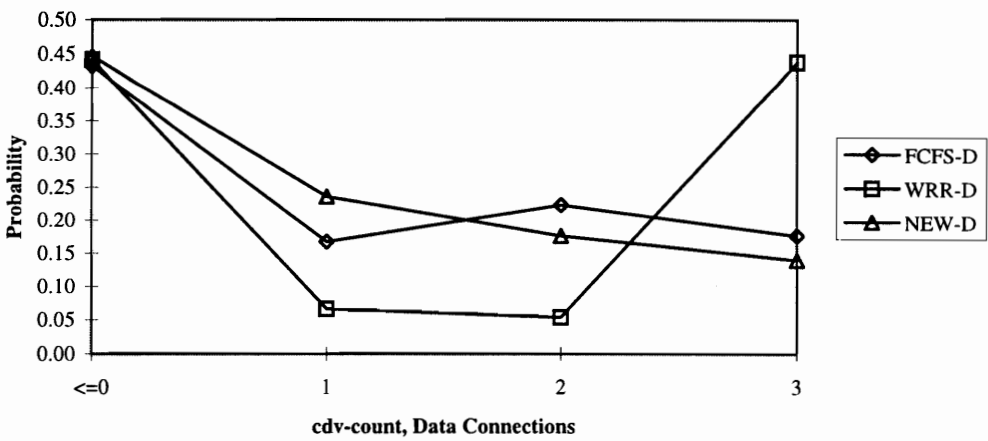
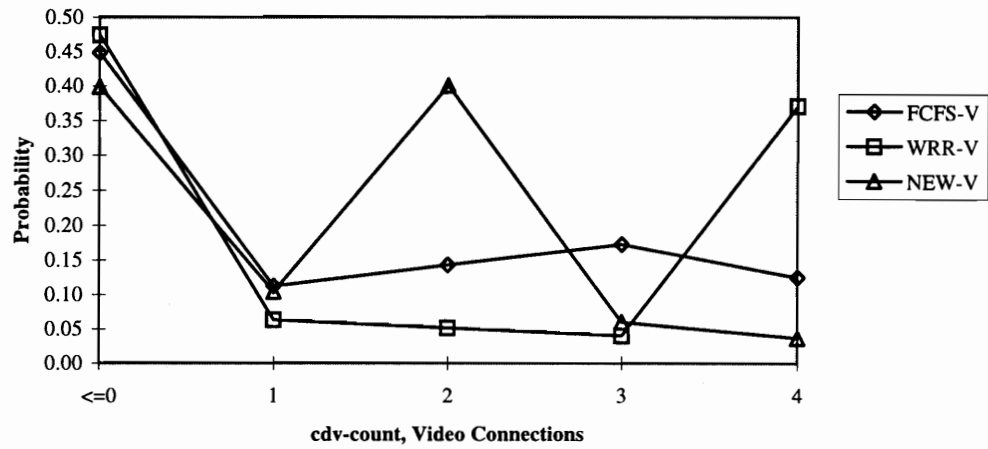
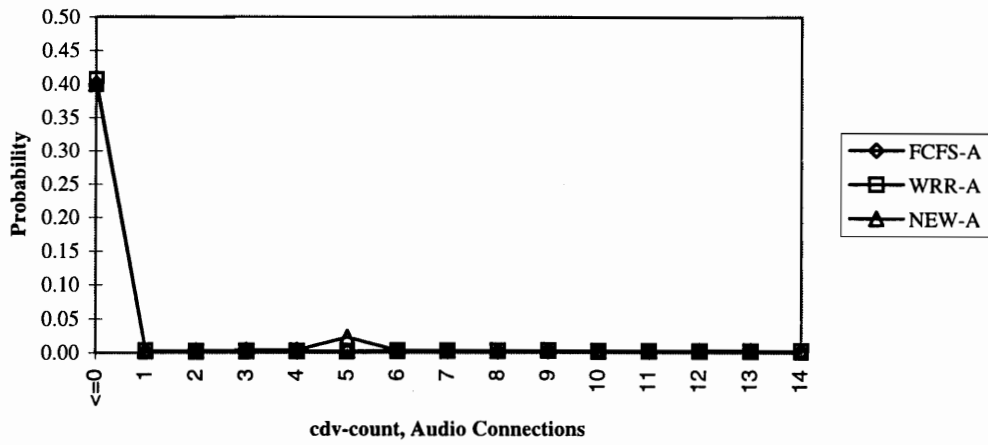


Figure 5.11 Simulation Results, Run 3A, cdv-count Distributions

Under the FCFS policy, the audio and video connections have similar wait time distributions, while the data connection looks completely different. As it turns out, all cells under FCFS have their wait times within the specified MPD's. There is a dip at the wait time of zero for both audio and video connections, which indicates that the cells in the audio and video connections expect to see at least one cell (most likely from the data connection) in the buffer when they arrive. The cells from all three connections have a probability of approximately 0.40 to have a cdv-count value of 0, with the remaining probabilities spread out fairly evenly among the possible cdv-count values.

The WRR policy protects connections with high transmission rates by scanning them more often, while it penalizes connections with low transmission rates. Since the transmission rates used for the video and data connections are fairly close, their wait time distributions are also fairly similar. The wait time distribution for the audio connection, on the other hand, suffers greatly. The probability of cells with a wait time of zero is less than 0.20, while for both video and data connections the corresponding value is above 0.40; the probability becomes smaller as the wait time becomes larger for all three connections. The cdv-count distribution for the audio connection matches fairly closely with the other two policies. The video and the data connections, on the other hand, have remarkably high probabilities at large cdv-count values.

The new algorithm performs as expected. The data connection has the largest transmission rate, however, due to its loose delay requirement (i.e., large MPD value), its wait time distribution is more spread out. Both the video and audio connections have stringent delay requirements, and the results show that both connections have high probabilities for small wait time values. As for the cdv-count distributions, all three connections show the characteristic "humps" at the border of the ACDV-window.

To better compare the performance of the three policies, the probabilities that cells are compliant with the parameters specified are calculated and listed in Table 5.16. Clearly, the new algorithm performs better than or at least the same as WRR for all connections with respect to both wait time and cdv-count. The new algorithm does not perform as well as does FCFS for wait time compliance for the audio connection; otherwise, it either outperforms or equals the performance of FCFS. The better performance of FCFS policy with the audio connection is obtained at the price of worse performance with the video connection. In terms of difference in probability, the new algorithm is better at protecting the video connection by about 0.180 while FCFS policy is better at protecting the audio connection by about 0.094. Special attention should be paid to the cell delay variation. The new algorithm offers a much better control than either FCFS or WRR for the audio and video connections, which have extremely strict ACDV requirements.

Table 5.16 Results for the Probability that Cells are Compliant with the Traffic Contract

		FCFS	WRR	NEW
Wait Time	Audio	0.696076	0.334237	0.602205
	Video	0.704699	0.744582	0.885347
	Data	1.000000	1.000000	1.000000
cdv-count	Audio	0.413174	0.418933	0.435573
	Video	0.702661	0.588765	0.903069
	Data	1.000000	1.000000	1.000000

5.5. Performance Comparison of Discard and Buffer Management Policies and Overflow Discard Policy with Fixed-Size Buffers

Run 4A and Run 4B compare the performance of the discard policy and the buffer management policy of the new algorithm with an overflow discard policy with a fixed-size buffer policy. A three-connection system is used in Run 4A and Run 4B, the total buffer size used is eleven, $c_1 = 2$ and $c_2 = 200$. The other parameters used are listed in Table 5.17.

Table 5.17 Parameters Used to Compare the Performance of Discard and Buffer Management Policies

Run	Buffer Size	Connection	λ	MPD	T	ACDV	ACLR
4A, 4B	11	1	0.125	10	16	5	0.000001
		2	0.250	10	8	5	0.000010
		3	0.500	10	2	5	0.000100

In the overflow discard policy, the size of a buffer for each connection is determined based on the ratio of the specified transmission rate of a connection over the sum of all connections' specified transmission rates. The specified transmission rate is simply the inverse of the peak emission interval. Therefore, the size of buffer for each connection under the overflow discard policy is allocated based on the proportion of specified transmission rates required by the connection with respect to all connections, and the buffer size is assumed to be fixed during a simulation run. Based on the values provided in Table 5.17, the buffer space allocated for connections 1, 2, and 3 under the overflow discard policy is 1, 2, and 8, respectively. Also, for connections 1 and 2, the actual transmission rates (λ 's) used are higher than the specified rates, i.e., the T's specified are only one half of the actual transmission rates, or the actual transmission rates are twice as fast

as the specified ones. The results of the simulation runs are tabulated in Table 5.18. The numbers in Table 5.18 represent the number of cells discarded for each connection in a replication.

From Table 5.18, it is clear that the discard and the buffer management policies of the new algorithm are superior to the overflow discard policy with fixed-size buffers. The results in Table 5.18 illustrate the strengths of the combined effect of the discard and the buffer management policies of the new algorithm: fairness in assigning loss among connections, and better utilization of system resources. The fairness of the new algorithm is demonstrated by the much higher but roughly equal proportion of losses assigned to connections 1 and 2 for violating the specified transmission rates, as shown by the values in the Actual Cell Loss Ratio row for Run 4B. On the other hand, connection 1 under the overflow discard policy suffers a greater amount of cell loss because of its smaller buffer size, even though the proportion of violation in transmission rates for both connections 1 and 2 are the same. Also, the system resources are better utilized under the new algorithm, as shown by the much lower overall cell loss (the Total Cell Discarded rows in Runs 4A and 4B) when compared to that of the overflow discard policy with fixed-size buffers.

Table 5.18 Simulation Results, Run 4A and Run 4B

		Connection 1	Connection 2	Connection 3
Run 4A	Overflow	126.033	68.933	0.000
	Total Cell Discarded	126.033	68.933	0.000
	Total Cell Transmitted	443.300	1061.000	2266.133
	Actual Cell Loss Ratio	0.284	0.065	0.000
Run 4B	MGCRA	1.267	2.167	2.500
	Rotation	0.000	0.000	0.000
	Total Cell Discarded	1.267	2.167	2.500
	Total Cell Transmitted	568.200	1127.733	2263.667
	Actual Cell Loss Ratio	0.002	0.002	0.001

5.6. Summary

In this chapter, the behavior of the new algorithm is studied via simulations. The effect of individual connection characteristic parameters on the new algorithm is analyzed in Section 2; the simulation result of the two-stage approach of the discard policy is shown in Section 3; Section 4 presents a comparative analysis of the first come first serve policy, weighted round-robin policy, and the new algorithm; and

Section 5 demonstrates the advantages of the combined effect of the discard and buffer management policies over the overflow discard policy with bandwidth-based fixed-size buffers. The new algorithm offers a lot of flexibility in controlling the behavior of an ATM node. The system variables, such as c_1 , c_2 , allow system providers to modify the algorithm's behavior to suit their needs. The main objectives of the new algorithm, which is to find a good compromise between delay and cell delay variations generated by a node's transmission policy and be fair in assigning losses among connections, are achieved, as demonstrated in Sections 4 and 5.

CHAPTER 6. CONCLUSIONS

This research has proposed and studied a new integrated algorithm to be used in ATM networks. The new algorithm has three components: the buffer management policy, the discard policy, and the transmission policy. The buffer management policy is adopted from papers presented by Chao, and is not analyzed in this research.

The discard policy employs a two-stage approach: the first stage enforces the source rate regulation by performing a modified GCRA (MGCRA) if buffer overflow is detected; the second stage discards cells rotationally based on rotation indices so that the average proportions of cell loss among connections relative to the specified ACLR's are about the same. The discard policy has the following advantages:

1. No cell is discarded if buffer space is available, regardless of the number of buffer spaces a connection takes; thus the system resources are better utilized.
2. All connections are protected by the MGCRA. If an imminent buffer overflow is detected, the MGCRA is activated to determine the connections (among the arriving cells) which are transmitting at faster rates than expected. Thus the connections that are compliant with their traffic contracts are protected.
3. If buffer overflow still occurs after the MGCRA, cells are discarded based on the rotation indices. The rotation indices are calculated from a connection's ACLR and PCR. The goal of discarding by rotation indices is that the long-term proportions of the actual cell loss ratios mimic that of the specified ACLR's. If a node becomes congested, all connections suffer in equal proportions, thereby fairness in loss among connections can be achieved.

The transmission policy tries to find a compromise between minimizing the cdv-count and minimizing the wait time of all connections. The policy accomplishes the objective by incorporating the time-varying priority concept. A set of priority indices is given to connections based on the wait time and cdv-count of cells at the heads of connections and connections' parameters from traffic contracts. The cell with the smallest priority index value is transmitted. The transmission policy has the following advantages:

1. The connection selection process adapts dynamically to the current status of all connections. Due to the diverse traffic distributions that the connections can have, the transmission policy can adjust its selection based on each connection's requirements.
2. As demonstrated in Chapter 5, depending on the cost factors used, the transmission policy can find a good compromise between two conflicting objectives of minimizing the wait time and cdv-count. The two other popular transmission policies, the first-come-first-served policy and the weighted round-robin policy, do not perform as well in general.
3. The transmission policy is extremely flexible. Weights and cost factors are used in the calculations of priority indices. The relative values of these numbers can control the behavior of the transmission policy, thereby offering system administrators options in fine tuning local system's behavior.

Several assumptions were made in this research to simplify the theoretical analysis. These assumptions may be varied in the future research to gain better understanding of the new algorithm. The following changes are suggested:

1. Use of different distributions for the arrival process. In this research, the arrivals were modeled by Bernoulli distribution. The Bernoulli distribution is good in modeling connections for which the probability of a cell being present in any time slot is equal. However, for many applications, different distributions should be used. For example, the on-off modulated Markov chain may be a good distribution to be used for bursty connections.
2. A large-scale simulation study. In this research, only the one-node behavior is studied. It would be interesting to perform a large-scale simulation experiment, with the new algorithm implemented in a network involving large number of nodes with diverse connections, to determine how the new algorithm would perform in terms of meeting the end-to-end requirements.

REFERENCES

- [1] Feng, T. -Y., "A Survey of Interconnection Networks," *Computer*, Vol. 14, No. 12, Dec. 1981, pp. 12-27.
- [2] Turner, J. S.; Wyatt, L. F., "A Packet Network Architecture for Integrated Services," *GLOBECOM '83, Conf. Rec.*, Nov. 1983, pp. 45-50.
- [3] Dias, D. M.; Kumar, M., "Packet Switching in N LOG N Multistage Networks," *GLOBECOM '84, Conf. Rec.*, Nov. 1984, pp. 114-121.
- [4] McMillen, R. J., "A Survey of Interconnection Networks," *GLOBECOM '84 Conf. Rec.*, Nov. 1984, pp. 105-113.
- [5] Turner, J. S., "New Directions in Communications (or Which Way to the Information Age?)," *IEEE Communications Magazine*, Vol. 24, No. 10, Oct. 1986, pp. 8-15.
- [6] Lea, C.-T., "What Should Be the Goal for ATM," *IEEE Network*, Vol. 6, No. 5, Sept. 1992, pp. 60-66.
- [7] McDysan, D. E.; Spohn, D. L., *ATM, Theory and Application*, McGraw-Hill, 1994, pp. 87-88.
- [8] Lin, A. Y.-M., Silvester, J.A., "On the Performance of an ATM Switch with Multichannel Transmission Groups", *IEEE Transactions on Communications*, Vol. 41, No. 5, May 1993, pp. 760 - 770.
- [9] *ibid.* McDysan, Spohn, pp. 190-191.
- [10] *ibid.* McDysan, Spohn, p. 239.
- [11] The ATM Forum, *ATM User-Network Interface Specification, Version 3.0*, PTR Prentice Hall, 1993.
- [12] Pattavina, A., "Nonblocking Architectures for ATM Switching," *IEEE Communications Magazine*, Vol. 31, No. 2, Feb. 1993, pp. 38-48.
- [13] *ibid.* McDysan, Spohn, pp. 280-283.
- [14] Karol, M.J.; Hluchyj, M.G.; Morgan, S.P., "Input Versus Output Queueing on a Space-Division Packet Switch," *IEEE Transactions on Communications*, Vol. COM-35, No. 12, Dec. 1987, pp. 1347-1356.
- [15] Hluchyj, M.G.; Karol, M.J., "Queueing in High-Performance Packet Switching," *IEEE Journal On Selected Areas In Communications*, Vol. 6, No. 9, Dec. 1988, pp. 1587-1597.
- [16] Liew, S.C., "Performance of Various Input-buffered and Output-buffered ATM Switch Design Principles under Bursty Traffic: Simulation Study," *IEEE Transactions on Communications*, Vol. 42, No. 2/3/4, Feb./Mar./Apr. 1994, pp. 1371-1379.
- [17] *ibid.* The ATM Forum, pp. 312-329.
- [18] *ibid.* The ATM Forum, pp. 77-79.

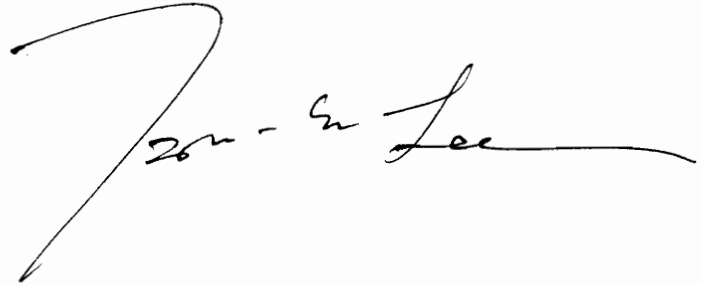
- [19] Woodruff, G.M.; Rogers, R.G.H.; Richards, P.S., "A Congestion Control Framework For High-Speed Integrated Packetized Transport," *IEEE GLOBECOM '88*, 1988, pp. 203-207.
- [20] Eckberg, A.E. Jr.; Luan, D.T.; Lucantoni, D.M., "Meeting The Challenge: Congestion And Flow Control Strategies For Broadband Information Transport," *IEEE GLOBECOM '89*, 1989, pp. 1769-1773.
- [21] Sidi, M.; Liu, W.-Z.; Cidon, I.; Gopal, I., "Congestion Control Through Input Rate Regulation," *IEEE GLOBECOM '89*, 1989, pp. 1764-1769.
- [22] Berger, A.W., "Performance Analysis of A Rate Control Throttle Where Tokens And Jobs Queue," *IEEE INFOCOM '90*, 1990, pp. 30-38.
- [23] Bala, K.; Cidon, I.; Sohraby, K., "Congestion Control For High Speed Packet Switched Networks," *IEEE INFOCOM '90*, 1990, pp. 520-526.
- [24] Ahmadi, H.; Guérin, R.; Sohraby, K., "Analysis of Leaky Bucket Access Control Mechanism with Batch Arrival Process," *IEEE GLOBECOM '90*, 1990, pp. 344-349.
- [25] Sohraby, K.; Sidi, M., "On the Performance of Bursty and Modulated Sources Subject to Leaky Bucket Rate-Based Access Control Schemes," *IEEE Transactions on Communications*, Vol. 42, No. 2/3/4, Feb./Mar./Apr., 1994, pp. 477-487.
- [26] Wu, X.; Lambadaris, I.; Lee, H.; Kaye, A. R., "A Comparative Study of Some Leaky Bucket Network Access Schemes," *Proceedings of The 1994 IEEE International Conference On Communications*, New Orleans, LA., 1994, pp. 1586-1591.
- [27] Kuang, L., "On The Variance Reduction Property of The Buffered Leaky Bucket," *IEEE Transactions On Communications*, Vol. 42, No. 9, Sept. 1994, pp. 2670-2671.
- [28] Liu, Z.; Towsley, D., "Burst Reduction Properties of Rate-Control Throttles: Departure Process," *Annals of Operations Research*, Vol. 49, 1994, pp. 51-78.
- [29] Anantharam, V.; Konstantopoulos, T., "Burst Reduction Properties of the Leaky Bucket Flow Control Scheme in ATM Networks," *IEEE Transactions on Communications*, Vol. 42, No. 12, Dec. 1994, pp. 3085-3089.
- [30] ibid. McDysan, Spohn, pp. 374-378.
- [31] Niestegge, G., "The Impact of Traffic Shaping on Broadband Network Performance," *AEÜ*, Vol. 47, No. 5/6, 1993, pp. 426-434.
- [32] Golestani, S. J., "Congestion-Free Transmission of Real-Time Traffic in Packet Networks," *IEEE INFOCOM '90*, 1990, pp. 527-536.
- [33] Trajkovic, L.; Halfin, S., "Buffer Requirements in ATM Networks With Leaky Buckets," *IEEE International Conference on Communications, Conference Records*, 1994, pp. 1616-1620.

- [34] Petr, D. W.; Frost, V. S., "Nested Threshold Cell Discarding For ATM Overload Control: Optimization Under Cell Loss Constraints," *IEEE INFOCOM '91*, 1991, pp. 1403-1412.
- [35] Yegani, P.; Krunz, M.; Hughes, H., "Congestion Control Schemes in Prioritized ATM Networks," *IEEE International Conference on Communications, Conference Records*, 1994, pp. 1169-1173.
- [36] Bemmell, V.; Ilyas, M., "A Unified Congestion Control Strategy in ATM Networks," *IEEE International Conference on Communications, Conference Records*, 1994, pp. 1600-1604.
- [37] Lin, A. Y.-M.; Silvester, J. A., "Priority Queueing Strategies and Buffer Allocation Protocols for Traffic Control at an ATM Integrated Broadband Switching System," *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 9, Dec. 1991, pp. 1524-1536.
- [38] Cheng, L.; Hughes, H.; Yegani, P., "Priority Scheduling in an ATM Network," *Proceedings of the 1994 Summer Computer Simulation Conference*, 1994, pp. 169-174.
- [39] Peha, J. M.; Tobagi, F. A., "Implementation Strategies for Scheduling Algorithms in Integrated-Services Packet-Switched Networks," *IEEE GLOBECOM '91*, 1991, pp. 1733-1740.
- [40] Chipalkatti, R.; Kurose, J. F.; Towsley, D., "Scheduling Policies for Real-Time and Non-Real-Time Traffic in a Statistical Multiplexer," *IEEE INFOCOM '89*, 1989, pp. 774-783.
- [41] Badr, H. G.; Mitrani, I.; Spirn, J. R., "An Adaptive-Priority Queue," *Applied Probability - Computer Science: The Interface*, ORSA-TIMS, 1982, Vol. 2, pp. 345-71.
- [42] Peha, J.M.; Tobagi, F. A., "A Cost-Based Scheduling Algorithm to Support Integrated Services," *IEEE INFOCOM '91*, 1991, pp. 741-753.
- [43] Chao, H. J.; Uzun, N., "An ATM Queue Manager with Multiple Delay and Loss Priorities," *IEEE GLOBECOM '92*, Orlando, Florida, Dec. 1992, pp. 308-313.
- [44] Demers, A.; Shenker, S.; Kashav, S., "Analysis and Simulation of a Fair Queueing Algorithm," *ACM SIGCOMM '89*, 1989, pp. 1-12.
- [45] Lim, Y.; Kobza, J. E., "Analysis of a Delay-Dependent Priority Discipline in an Integrated Multiclass Traffic Fast Packet Switch," *IEEE Transactions on Communications*, Vol. 38, No. 5, May 1990, pp. 659-665.
- [46] Ramaswami, V.; Lucantoni, D. M., "Discussant's Report on 'Algorithmic Analysis of a Dynamic Priority Queue,'" *Applied Probability - Computer Science: The Interface*, ORSA-TIMS, 1982, Vol. 2, pp. 205-206.
- [47] Karol, M. J.; Schwartz, S. C., "Multiple-Access Protocols: Fairness in Heterogeneous Systems," *IEEE Transactions on Communications*, Vol. 42, No. 6, June 1994, pp. 2276-2281.
- [48] Hahne, E. L., "Round-Robin Scheduling for Max-Min Fairness in Data Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 7, Sept. 1991, pp. 1024-1039.

- [49] Greenberg, A. G.; Madras, N., "How fair is fair queueing?" *Journal of the Association for Computing Machinery*, Vol. 39, No. 3, July 1992, pp. 568-598.
- [50] Katevenis, M.; Sidiropoulos, S.; Courcoubetis, C., "Weighted Round-Robin Cell Multiplexing in a General-Purpose ATM Switch Chip," *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 8, Oct. 1991, pp. 1265-1279.
- [51] Wang, Y. -T.; Lin, T. -P.; Gan, K. -C., "An Improved Scheduling Algorithm for Weighted Round-Robin Cell Multiplexing in an ATM Switch," *IEEE International Conference on Communications, Conference Records*, 1994, pp. 1032-1037.
- [52] Saran, H.; Keshav, S.; Kalmanek, C. K.; Morgan, S. P., "A Scheduling Discipline and Admission Control Policy for Xunet 2," *NOSSDAV '93*, 1993, pp. 89-101.
- [53] Panwar, S. S.; Philips, T. K.; Chen, M.-S., "Golden Ratio Scheduling for Flow Control with Low Buffer Requirements," *IEEE Transactions on Communications*, Vol. 40, No. 4, Apr. 1992, pp. 765-772.
- [54] Zhang, L., "VirtualClock: A New Traffic Control Algorithm for Packet-Switched Networks," *ACM Transactions on Computer Systems*, Vol. 9, No. 2, May 1991, pp. 101-124.
- [55] Chao, H. J., "A Novel Architecture for Queue Management in the ATM Network," *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 7, Sept. 1991, pp. 1110-1118.
- [56] Chao, H. J.; Pekcan, I. H., "Queue Management with Multiple Delay and Loss Priorities for ATM Switches," *IEEE International Conference on Communications, Conference Records*, 1994, pp. 1184-1189.
- [57] Golestani, S. J., "Congestion-Free Communication in Broadband Packet Networks," *IEEE ICC '90*, 1990, pp. 489-494.
- [58] Law, A. M.; Kelton, W. D., *Simulation Modelling & Analysis*, Second Edition, McGraw-Hill, Inc., 1991.

VITA

Mr. Tzu-En Lee was born on the December 26, 1967 in Taipei, Taiwan. He acquired his bachelor's degree in Electrical Engineering from the Virginia Polytechnic Institute and State University in 1991 and received the Master of Science in Engineering degree in Systems Engineering from the University of Pennsylvania in 1992. He worked with Dr. Sarin from the Department of Industrial and Systems Engineering at Virginia Polytechnic Institute and State University for a Master of Science degree in Industrial and Systems Engineering. Mr. Lee is going to work at Systems Research and Applications Corporations starting September 1995.

A handwritten signature in black ink, reading "Tzu-En Lee". The signature is written in a cursive style with a large, sweeping initial "T" and a long horizontal line extending from the end of the name.