This is to certify that the

dissertation entitled

DESIGN AND ANALYSIS OF A BUFFER MANAGEMENT SCHEME FOR MULTIMEDIA TRAFFIC IN ATM NETWORKS

presented by

MARWAN M. KRUNZ

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in Electrical Eng.

Hermann D. Hughes
Major professor

Date 7-18-95

DESIGN AND ANALYSIS OF A BUFFER MANAGEMENT SCHEME FOR MULTIMEDIA TRAFFIC IN ATM NETWORKS

By

Marwan M. Krunz

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering

1995

ABSTRACT

DESIGN AND ANALYSIS OF A BUFFER MANAGEMENT SCHEME FOR MULTIMEDIA TRAFFIC IN ATM NETWORKS

By

Marwan M. Krunz

This research is concerned with the design and analysis of a buffer management scheme to be implemented in the switching and multiplexing nodes of a broadband network such as *Broadband Integrated Services Digital Network* (B-ISDN). The goal of B-ISDN is to provide a unified means of communications for many types of applications which generate data, voice, video streams, and combinations of these streams (i.e., multimedia sources). Such a goal has been made possible through the standardization of the *Asynchronous Transfer Mode* (ATM) protocol. One of the most challenging issues in the design of B-ISDN/ATM networks is to guarantee the performance requirements (known as *Quality-of-Service* (QoS)) of many traffic sources that differ in their statistical characteristics. Since these streams also differ in their cell loss and delay sensitivities, the network must be designed to support multiple levels of loss and delay performance.

In this work, we introduce a new approach to providing QoS guarantees based on buffer management. In this approach, loss and delay priority mechanisms are both used in the design of a buffer management scheme that supports multiple levels of QoS requirements. The proposed scheme is referred to as *Nested Threshold Cell Discarding with Multiple Buffers* (NTCD/MB). We describe the general structure of NTCD/MB and demonstrate its effectiveness via analysis and simulation. NTCD/MB is found particularly useful when the aggregate traffic arriving at an ATM node consists of several heterogeneous streams.

We study the performance of NTCD/MB under different types of traffic, including bursty traffic streams (e.g., data sources) and *variable-bit-rate* (VBR) video streams. Accurate models will be used to characterize the traffic sources under study. Some of these models are already being used in the literature, such as the fluid model and the Markov-modulated Poisson process model. Additionally, we propose new traffic models to characterize the VBR video streams that are generated based on the emerging Moving Picture Expert Group (MPEG) compression standard. Performance evaluation of the NTCD/MB is used in the dimensioning and partitioning of the buffer space, so that multiple levels of loss and delay performance requirements can be supported by the network.

To my wife, Carine

ACKNOWLEDGMENTS

Achievements are meaningless if no one else shares in them. This pinnacle of my efforts would not have been possible without the assistance and contributions of several people.

Foremost among those whom I thank is my advisor, Dr. Herman Hughes, for his continuous support and guidance during the course of this research. I owe him no less gratitude for the manner in which he has overseen my career as a graduate student. I would like to thank the members of my guidance committee: Dr. David Fisher for his editorial, technical, and thematic comments on this dissertation; Dr. Diane Rover for providing me constantly with valuable advice on my career goals and also for her thought-provoking questions and comments; Dr. Vince Melfi for sharing his valuable insights in probability theory; and Dr. Philip Mckinley for helping me understand the basic concepts of computer networks.

This dissertation could not have been completed if it were not for the generosity and patience of my beloved wife. Through her love and support I overcame many times of hardship and frustration. I must also extend my thanks and appreciation to my family who continuously provided me with their love and support. Many thanks to my friends Mohannad Bataineh, Ahmad Al-Ostaz, and Eman El-Sheikh for providing me with many valuable comments for my oral presentation.

This research was supported by a grant from the National Science Foundation (# NCR 9305122).

TABLE OF CONTENTS

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Scope and General Goals

This research is concerned with the design and analysis of a buffer management scheme for multimedia traffic in a Broadband-Integrated Services Digital Network (B-ISDN). Buffer management is an important aspect in the design of a switching or multiplexing B-ISDN node. The goal of B-ISDN is to provide a unified means of communications for a wide range of applications including video-telephony, multi-spectral imaging, High-Definition TV, medical imaging, multimedia conferencing, and LAN interconnections. Achieving this goal became possible through the standardization of the Asynchronous Transfer Mode (ATM) transport protocol [10, 55]. ATM is an attractive switching technology which is characterized by high-speed fiber transmission facilities and simple hardwired network protocols. These protocols are designed to match the transmission speeds of communication links. In ATM, data generated by various traffic sources are encapsulated into fixed-length packets known as *cells*. The size of an ATM cell is 53 bytes, of which 5 bytes are used as a header. As a backbone switching network, ATM is designed to minimize the overhead incurred in processing network protocols. The switching in an ATM network is done in hardware ATM

switches, unlike traditional packet-switched networks in which switching is done by computer running communication processes.

One of the most challenging issues in the design of B-ISDN/ATM networks is to provide *guaranteed* service to diverse traffic streams without underutilizing the available bandwidth capacity. These performance guarantees, which are known as the *Quality-of-Service* (QoS) requirements, are particularly important for time-critical applications such as interactive video and voice communications. Common measures for the QoS requirements are the cell loss rate, the absolute end-to-end cell delay, and cell delay variation (i.e., jitter). Other measures include the cell loss rate within a burst and the call blocking probability. The notion of guaranteed service is a major migration from the design methodology of conventional packet networks that provides either a best-effort delivery or a reliable transport delivery (e.g., TCP/IP transport technology). The new types of traffic that will be transported over B-ISDN require more than just reliability; they require guarantees on the delivery performance perceived by their cells. In packetized voice, for example, each voice packet (or cell), must be delivered in no more than 200 msec so that this packet can be decoded and played back in real-time.

It should be noted, however, that providing performance guarantees in ATM networks does not necessarily imply that a fixed amount of bandwidth must be exclusively reserved for a given connection (as in the case of circuit-switched networks which are based on time-division multiplexing (TDM)). Instead of using TDM, ATM networks employ *statistical multiplexing* to efficiently utilize network resources (i.e., bandwidth and buffers). In statistical multiplexing cells from various connections share the network resources on a need basis. Statistical multiplexing is particularly useful when the aggregate traffic consists of a number of sources each of which is characterized by alternating active and idle periods, with the active periods being shorter (on the average) than the idle periods [70].

Ideally, the network must guarantee the QoS requirements of all communicating sessions while, simultaneously, taking advantage of statistical multiplexing. Since traffic sources generally differ in their QoS requirements, the network is being designed to provide multiple grades of service. Each grade of service guarantees specific (predetermined) levels of loss and delay performance. Designing the network to guarantee multiple grades of service while employing statistical multiplexing poses difficult challenges concerning the proper buffer management and bandwidth allocation schemes to be implemented at ATM nodes. In certain cases, the loss of some bandwidth by non-statistical (i.e., synchronous) multiplexing could be justified by the need to provide QoS guarantees for certain types of traffic. A combination of statistical and non-statistical multiplexing may result in moderate bandwidth utilization and simple traffic control [70].

Performance guarantees are classified as *deterministic* and *statistical* [69]. Deterministic guarantees provide bounds on the performance experienced by all cells within a session. Statistical guarantees ensure that not more than a certain fraction of the cells within a session will experience a performance worse than some specified value. Statistical guarantees are often provided asymptotically (i.e., over an infinite time interval), but can also be given over fixed time intervals. Three general approaches to providing deterministic and statistical QoS guarantees can be identified [44]:

1. *The tightly controlled approach* – This approach provides deterministic guarantees by sacrificing some bandwidth to preserve the characteristics of a traffic stream when this stream traverses multiple nodes. In addition to low utilization, this approach results in increasing the implementation complexity of buffer management.

2. *The approximate approach* – Here, traffic models are used to analyze the queueing behavior of the traffic and to provide approximate statistical guarantees. This approach depends mainly on queueing analysis of ATM networks. Com-

plex nontrivial models are being used to characterize the stochastic behavior of the traffic. Such models take into consideration the correlations between arrivals and the variability of the arrival rate of a given stream [12, 22, 52, 56].

3. *The bounding approach* – Bounds on performance can be developed to provide statistical and deterministic guarantees. No assumptions are being made about the stochastic behavior of the traffic. This approach has been used to provide asymptotic stochastic bounds on performance, as well as bounds that hold over fixed time intervals.

Many factors influence the choice of a particular approach to guaranteeing the QoS requirements. These factors include the nature of the requested QoS (e.g., cell loss rate over finite duration versus infinite duration), the complexity of the admitted traffic streams (e.g., correlations between cell interarrival times), and the interactions among different streams at various multiplexing and switching nodes. Hence, it is not always possible to analytically evaluate the performance of the network and provide QoS guarantees using a given approach.

In this research, we adopt the approximate statistical approach to guaranteeing the QoS requirements. This approach is favored over other approaches because it permits us to derive many tractable performance results, and consequently, provide closed-form expressions for several measures of performance (e.g., cell loss rates and mean queueing delay). Our emphasis here is on the performance guaranteed by a single ATM node (a multiplexer or a switch). Since the user QoS requirements are requested on an end-to-end basis, it is the responsibility of the service provider (i.e., the network) to assign various proportions of these requirements to individual nodes along the path of the connection. If each of these nodes can guarantee its proportion, then the end-to-end QoS requirements are consequently guaranteed. Throughout this thesis, the QoS will be used to refer to the fraction of the end-to-end QoS that has been assigned to a single node.

The general goal of this research is to introduce a new approach to providing QoS guarantees based on buffer management. In this approach, loss and delay priority mechanisms are both used in the design of a buffer management scheme that supports multiple levels of cell loss and cell delay performance. We propose such a scheme and refer to it as *Nested Threshold Cell Discarding with Multiple Buffers* (NTCD/MB)). As will be demonstrated later, NTCD/MB is most appropriate in heterogeneous and multimedia traffic environments, where multiple grades of service are needed. Our objectives can be summarized as follows:

1. Introduce the NTCD/MB scheme and demonstrate its effectiveness in guaranteeing multiple levels of loss and delay performance.

2. Evaluate the performance of NTCD/MB under different types of traffic, including data, voice, and video sources.

3. Provide "optimal" dimensioning and partitioning for the buffer space associated with the proposed scheme.

1.2 Contributions of the Thesis

The three objectives listed above have been addressed throughout the course of this research. A general structure for the NTCD/MB buffer management scheme was developed and is outlined at the end of *Chapter 2*. Because of the merits of this scheme (with regard to performance and implementation complexity), we propose its implementation in ATM switches and multiplexers.

Analysis and simulations were used to study the performance of NTCD/MB. Our focus was on particular forms of this mechanism that are useful for realistic traffic mixes. Simulations of the NTCD/MB (*Chapter 3*) were conducted to evaluate the performance of the NTCD/MB under data and voice traffic inputs. In these simula-

tions, the aggregate traffic is modeled using the Markov-modulated Poisson process (MMPP) model. This complicated random process is appropriate in characterizing voice and data streams. Simulations of buffer systems become prohibitive when the values for buffer overflow probabilities are below 10^{-6}. The QoS requirements for many applications include a cell loss rate in the order of 10^{-10} (e.g., File Transfer Protocols (FTP)). In these cases, traffic control studies must rely on analytical results to predict and evaluate the queueing performance in a given buffer system.

We provide a complete fluid-flow analysis for the proposed NTCD/MB (*Chapter 4*). The fluid models used in this analytical study are appropriate in characterizing data and voice streams. In addition, our analysis can be easily extended to certain types of compressed video (based on the model of Maglaris *et al.* [52]). Our analytical results include the distribution for the buffer occupancy (for each buffer in the NTCD/MB mechanism), the waiting time distributions, the overflow probabilities, the throughputs, and other statistics. Based on these results, it was possible to predict the cell loss probability for each buffer in the range of 10^{-20}. These values are impossible to achieve via simulations.

Performance studies in *Chapter 3* and *4* are restricted to voice and data sources since traffic models for these sources have been developed and are widely accepted. On the other hand, models for compressed video traffic are still in the very early stages of development. This is mainly due to the complexity of video traffic and the variety of compression algorithms in use. Current models for video streams are only applicable to certain types of video (e.g., video-phones) and based on certain (non-standard) compression techniques. In 1991, the International Standards Organization (ISO) completed the first draft on a standard compression algorithm known as *Moving Picture Expert Group* (**MPEG**) [26]. This draft (referred to as MPEG-I) defines the compressed bit stream for video and audio at 1.5 Mbits/s bit rate. The standardization of MPEG makes it worthwhile to study the performance of NTCD/MB

under MPEG-compressed video streams. To conduct such a study, we needed a traffic model for an MPEG stream or, alternatively, actual traces of MPEG streams to be used in trace-driven simulations of NTCD/MB. Unfortunately, neither traffic models nor empirical data were available when our research was initiated. (mainly due to the short history of MPEG). For this reason, we extend the goals of our research to include the characterization and modeling of MPEG-compressed streams. The resulting models are to be used in studying the performance of NTCD/MB under video (as well as data and voice) traffic. Empirical studies were first conducted to capture, digitize, and MPEG-compress a long stream of video taken from an entertainment movie. We used these data to develop two models for an MPEG stream (*Chapter 5*). The first model is based on fitting the number of cells in MPEG frames by lognormal probability density functions (*pdf*s). These *pdf*s were used along with the compression pattern to generate synthetic streams based on this model. In addition to the lognormal fits and the compression pattern, the second model incorporates, as well, the scene dynamics in the movie. Both models proved to be sufficiently accurate in capturing several aspects of the actual traffic. More importantly, their queueing performance is quite similar to the trace-driven performance.

With traffic models for MPEG video streams in hand, we studied the performance of the NTCD/MB mechanism under data, voice, and MPEG video streams (*Chapter 6*). By combining our analytical and simulation results, we computed the optimal buffer-bandwidth pairs that are needed to guarantee given sets of QoS requirements. Although the treatment in *Chapter 6* was carried for a case study, it can be used, in general, for the dimensioning and optimization of resources in any structure of the NTCD/MB mechanism and under any sets of QoS requirements. Finally, the thesis is concluded in *Chapter 7*, and pointers to future research are also provided.

Chapter 2

Buffer Management and Priority Mechanisms

2.1 Introduction

Traffic streams transported over a fully evolved multimedia network such as B-ISDN are expected to have a wide range of performance requirements (i.e., QoS). Not only is this true for a heterogeneous mix of traffic, but also for certain individual traffic sources that generate cells with several loss and/or delay requirements, as in the case of subband coded video [35]. To satisfy the QoS requirements for all streams, the network may choose to provide indistinguishable transport service based on the most stringent QoS requirements. Such a strategy is too restrictive and significantly underutilizes network resources, particularly when the traffic streams with the most demanding requirements constitute a minor fraction of the total traffic. Alternatively, the network may offer multiple bearer capabilities (i.e., grades of service), resulting in efficient resource utilization. This requires assigning different levels of 'delivery' priority to incoming cells, and then offering differential service to these cells using priority queueing mechanisms. Such mechanisms can be implemented at various

buffering stages in the network. A priority mechanism in this context refers to the scheduling and/or admission algorithm in a buffer (queue), and is often called a *buffer management scheme.*

To maintain a simple and fast traffic control, it is reasonable to support a small number of priority classes. The use of priority gives the network the flexibility to dynamically adjust to different traffic mixes, resulting in an increase in the total admissible load as compared to non-prioritization [35]. Priority mechanisms are also useful in other areas of traffic control such as traffic policing. Policing is a preventive control mechanism which ensures that an admitted traffic stream conforms to its contract with the network. This contract normally specifies certain traffic parameters that characterize the behavior of the source (e.g., peak bit rate, mean bit rate, average burst length). If a stream violates its contract, the network provider can assign a lower priority to the violating cells and transport them with less performance guarantees (or with no guarantees at all).

Once priority classes have been defined, the network must be designed to map these classes into corresponding guaranteed grades of service. This requires a proper buffer management scheme to be implemented in the buffers of switching and multiplexing ATM nodes. Such a scheme ensures that the loss or delay performance experienced by a traffic stream is directly related to its class.

The study of priority queueing schemes (as to their role in providing guaranteed QoS in ATM networks) involves two main issues: one is the design of these schemes and the other is the evaluation of their performance. In designing a priority scheme for ATM networks, one must take into account the hardware requirements to implement such a scheme. These requirements could limit the practicality of the priority scheme. Evaluating the performance of a priority scheme is often more challenging than the design itself since it requires a considerable amount of research to provide a theoretical or an experimental assessment of performance. In this section, we review

the most relevant literature which addressed both issues. We further demonstrate the main drawback of current buffer management schemes, namely their inefficiency in providing multiple grades of service in both principal measures of performance: loss and delay. This motivates the introduction of a new buffer management scheme, which is the focus of this research.

2.2 Design of Buffer Management Schemes

Buffer management can be decomposed into two components: service discipline, which determines the order in which cells in the buffer are served; and buffer access control, which deals with admitting cells to the buffers [50]. Explicit or implicit priority rules may be applied to either or both disciplines. Accordingly, two types of priority queueing mechanisms can be identified, based on *where* the priority rule is enforced: *delay* and *loss* priority mechanisms.

2.2.1 Delay Priority Mechanisms

In a delay priority mechanism, the priority rule takes place at the output of the buffer when a cell is to be served. It is in essence a scheduling algorithm. Higher priority cells are given preferential service over lower priority cells in the scheduling order. Delay priority mechanisms are quite useful in the presence of time-critical traffic (for example, alarms and real-time control messages in manufacturing environments) that requires a rapid transfer of data across the network. Under certain traffic mixes, these mechanisms are also found to enhance the overall performance by reducing the cell loss rate in the network. Some of the popular delay priority mechanisms are as follows [9, 49]:

- Head-Of-the-Line (HOL).
- Queue Length Threshold (QLT).

- Minimum Laxity Threshold (MLT).
- Head-Of-the-Line with Priority Jumps (HOL-PJ).
- Earliest-Deadline-First (EDF).

The HOL scheme is a *static* priority scheduling policy which has been extensively studied in queueing theory books. It is static in the sense that high priority cells (assuming two classes for illustration purposes) are always scheduled to be served before all low priority cells, regardless of the traffic conditions and the relative buffer occupancy from the two classes. The high priority class is usually a delay-sensitive (e.g., real-time) traffic. The low-priority class contains all other types of traffic which are non-delay-sensitive, but possibly loss-sensitive. A non-preemptive priority is usually assumed. When the queue contains more than one high-priority cell, these cells are served in a FCFS fashion. This applies also to low priority cells when the queue does not contain high priority cells. HOL provides relatively low delays for the delay-sensitive traffic while causing high losses for other types of traffic. The performance of the low priority traffic is severely degraded if a large portion of the network traffic belongs to the high priority class. Better scheduling can be achieved using *dynamic* priority schemes such as QLT, MLT, and HOL-PJ. In these schemes, priority levels are dynamically adjusted to cope with the traffic conditions and the content of the buffer. Chipalkatti *et al.* [9] have shown that the QLT and MLT schemes offer better performance for the low priority traffic compared to HOL. The QLT scheme gives priority to the real-time traffic whenever the number of nonreal-time cells in the buffer reaches a certain threshold; otherwise the real-time traffic is given priority. In the MLT scheme, the scheduling priority is given to real-time traffic as long as the minimum laxity of real-time cells is less than some threshold (the laxity of a cell is defined as the time spent in queue before its deadline expires). Therefore, a real-time cell remains in the queue until either the cell is transmitted within its deadline or

discarded and considered lost. In this case the priority is given to the nonreal-time traffic. Both the MLT and QLT schemes can be used to achieve a desired level of performance for each traffic class by choosing an appropriate value for the threshold parameter. The QLT discipline, however, is easier to implement because it involves less processing at each switching node. Analytical studies of these disciplines were reported in [9].

The HOL-PJ scheme was proposed by Lim and Kobza [49] for an ATM switch serving multiple classes of delay-sensitive traffic. In this scheme, cells with different delay requirements (sometimes within a single traffic stream) enter one of several queues. Each queue is associated with a certain delay constraint, with tighter constraints having higher priority. For example, voice and video traffic are assigned a higher priority than other types of traffic, since their cells must be delivered no later than a predetermined play-out time. Each queue in the HOL-PJ scheme implements a FCFS discipline, with higher priority queues having non-preemptive priority over lower priority queues. A cell leaves its queue when its maximum queueing delay exceeds a predetermined limit, and it joins the end of the next higher priority queue. This process continues until the cell is served. Therefore, the maximum queueing delay for a given cell is determined by the sum of delay constraints at all the queues with priority levels equal to or higher than the starting priority level of the cell. The performance for each class can be controlled by adjusting the values of the delay limit. The processing overhead required to move cells between queues and the need for time stamps and time-outs are some undesirable features of this scheme.

Under the EDF scheme [8], the delays experienced by cells along long routes are reduced at the expense of longer delays for cells along short routes. One popular simplified version of EDF is called Older-Customer-First (OCF). In this policy, higher priority is given to older cells (based on time stamps); a new arriving cell joins a queue behind older cells and ahead of younger ones. Thus, the older cells should

expect shorter waiting times. OCF can be used, for example, in packetized voice communications to compensate for the variable delays incurred due to queueing of cells at previous nodes. This improves the quality of the speech by minimizing the delay variability of voice cells. A disadvantage of this scheme is the processing overhead required to move and monitor cells among queues. The practicality of this scheme is not clear yet and awaits future study. Several other delay-dependent priority disciplines have also been proposed (see [8] and references therein).

One major disadvantage of delay priority mechanisms is the need for a selection algorithm to determine the order in which cells are to be served. Such an algorithm significantly increases the implementation complexity of these schemes and limits the switching speed. Recent emphasis has focused on loss priority mechanisms because of their simplicity that is necessary to implement fast ATM switches.

2.2.2 Loss Priority Mechanisms

In this type of priority mechanisms, the priority rule is applied at the input to the buffer. Cells of higher classes have priority over cells of lower classes with regard to being admitted to the buffer. A loss priority mechanism is, therefore, a selective cell discarding scheme in which a cell of a given class is dropped (rather than delayed) to accommodate a higher priority cell in the buffer. Loss priority mechanisms were first introduced to control congestion in ATM nodes [3]. They are needed to protect an ATM node from the stochastic fluctuations in the traffic which may temporarily deplete network resources (nodal buffers and transmission links) and cause congestion to develop. These schemes are also used to guarantee different cell loss requirements of various classes of traffic. The use of a loss priority scheme results in an increase in the total admissible load (i.e., overall utilization) or, alternatively, decrease in the required buffer space.

Several loss priority schemes were proposed and analyzed [76, 59, 35, 34, 72, 50, 78].

The scheme studied by Yin *et al.* [76] is based on selectively discarding voice cells whose loss will produce the least degradation in the reconstructed voice signal. The results in [76] indicate that during congestion periods the average waiting time of data cells can be significantly reduced by discarding a small fraction of voice cells. An advantage of this mechanism is that it lends itself to simple hardware implementation. Petr *et al.* [59] used a similar technique to control temporal congestion in a combined voice-data network by taking advantage of the inherent structure of the speech signal. For instance, the relative importance of different segments of the speech is recognized, and active speech is given priority over background noise during silence periods. A minimum short-term signal degradation is deliberately introduced by dropping less important information in order to improve the overall network performance. An early example of this scheme is found in the older T1 multiplexer systems, in which the least significant bit of the PCM speech samples is periodically discarded to provide signaling information.

Congestion control in packetized voice systems using priority discarding was also proposed in [77] and [75]. In these systems, by using either an embedded coding technique [19] or even/odd sample method [31], the more significant information is carried by high priority cells, and the less significant information by low priority ones. It is obvious that some low priority cells experience fairly long waiting times and can not be included in reconstructing the voice signal. Therefore, it is appropriate to discard these cells at the transmitter in order to reduce congestion in the network.

In other studies [11, 57], a slightly different priority discarding scheme was proposed for voice traffic. More important bits (not cells) are given higher priority over less important ones. A possible disadvantage of this discipline is the excessive per-bit processing overhead, which makes it less attractive for the ATM environment.

Priority discarding mechanisms were also proposed for video traffic [32, 18]. An image that is encoded by a layered coding technique is divided into two segments: a

high priority segment that contains the important cells (e.g., synchronization information), and a low priority segment that consists of the ordinary video data. When congestion occurs, the low priority cells are dropped. A layered coding technique for video signals was introduced by Kishino *et al.* [32] which was shown to be suitable for cell loss compensation in ATM networks. A two-layer conditional replenishment coding technique of variable bit rate (VBR) video signals was proposed by Ghanbari [18], where the more important video cells (referred to as 'guaranteed' cells) are given priority over ordinary cells. The first layer of coding generates the guaranteed cells while the second layer produces enhancement (add-on) cells. Reconstruction of the picture is based primarily on high priority cells.

When a loss mechanism is being examined from a buffer access prospective, it is commonly referred to as a *space* priority mechanism. This is due to the fact that this mechanism deals with priorities regarding the utilization of the buffer space. One bit in the ATM cell header was reserved to implement space priorities. Policing functions and buffer access mechanisms may use this bit to mark cells based on a negotiated contract for bandwidth allocation. Therefore, violating cells are marked as low priority and are rejected from entering the network when congestion occurs. Cell marking may be done either at the call level or at the cell level.

Several space priority mechanisms have been proposed [20, 16, 25, 60, 71]. These schemes are variations of two basic priority mechanisms, which are described below.

2.2.2.1 The Push-Out Mechanism

In this mechanism [20], cells enter one shared buffer up to the maximum buffer size. If a high priority cell arrives at a saturated buffer that contains low priority cells, a low priority cell is dropped and its place is given to the high priority cell. Despite its efficiency, *push-out* requires a complicated buffer management to preserve the sequencing of cells.

2.2.2.2 The NTCD Mechanism

The Nested Threshold Cell Discarding (NTCD) mechanism [16, 25, 72, 40, 39, 60, 71, 12, 50, 59] was first introduced and analyzed by Garcia and Casals [16]. It is also known as the *partial buffer sharing*. Under NTCD (Fig. 2.1) the buffer is partitioned by a number of thresholds n that corresponds to the number of priority classes. Cells of priority class i enter the buffer up to threshold level T_i. If the buffer occupancy is above T_i, arriving cells of class i are dropped. NTCD results in a slightly less total admissible load compared to *push-out* [35]. The advantages of using this scheme for congestion control in the network are:

1. The NTCD mechanism combines good performance characteristics with a simple buffer management that can be implemented in hardware.

2. By adjusting the threshold values T_i $(i = 1, \ldots, n)$, the NTCD mechanism provides the flexibility to adapt an ATM node to different traffic mixes.

Compared to other mechanisms, NTCD appears to be the best candidate for loss-based congestion control, as it provides a good compromise between performance and complexity.

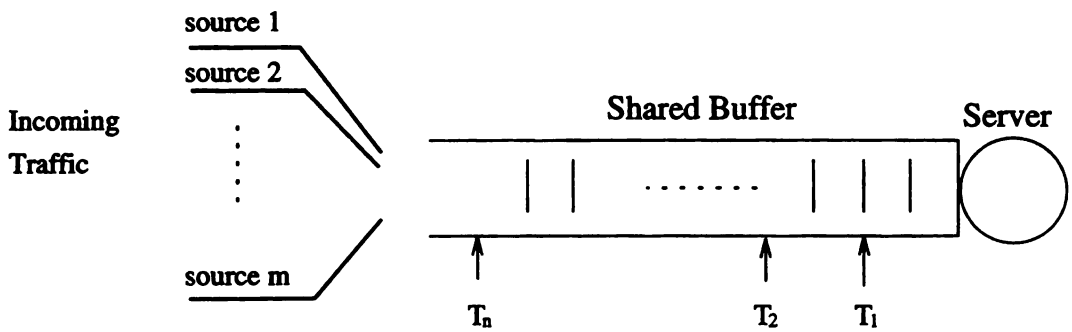

Figure 2.1: Structure of NTCD with one buffer.

2.3 Analysis of Buffer Management Schemes

Performance evaluation of buffer management schemes is essential to predict the overall network performance and consequently provide guaranteed QoS requirements to all traffic streams. It is also used for optimization and dimensioning of the network resources (i.e., buffer space and bandwidth). Ideally, performance evaluation should be based on measurements taken directly from an actual fully-operating ATM network. Since a such network does not exist yet, two other approaches to performance evaluation have been used by network analysts. The first (and most common) approach is based on a presumed traffic model that encapsulates some of the stochastic characteristics of the actual input stream(s). Such a model can be used in subsequent queueing analysis or simulations of buffers at ATM nodes. The second approach is based on traces of actual traffic streams. These traces can be used as traffic inputs to simulations (i.e., *trace-driven simulations*). Indeed, this latter approach relies heavily on the availability of such traces. Whatever assumptions are used to characterize the arrival process of the traffic will have a significant impact on the predicted performance. Therefore, it is necessary when studying the performance of buffer management schemes to use traffic models that capture the most important (with regard to queueing performance) characteristics of the actual traffic.

Before describing some of the traffic models which have been used in the literature for the analysis of ATM networks, we briefly discuss one important feature of the traffic that has a significant impact on performance, namely traffic correlations.

2.3.1 Burstiness and Traffic Correlations

The complexity of the traffic in B-ISDN is a natural consequence of integrating over a single communication channel a diverse range of traffic sources (i.e., video, voice, and data) that significantly differ in their traffic patterns as well as their performance

requirements. This heterogeneous traffic mix can not be adequately approximated by a simple Poisson process [65]. Specifically, bursty traffic patterns generated by data sources and VBR real-time applications (e.g., compressed video and audio) exhibit certain degrees of correlations between arrivals that preclude the use of renewal processes to model these patterns, not to mention a Poisson process. Even if the traffic generated by a single source was approximated by a renewal process, the aggregate traffic of the superposition of several sources is a complex non-renewal process which is modulated (i.e., controlled) by the number of active sources at each instant [22]. Correlations between arrivals have been found to cause a considerable degradation in the network performance (as measured by cell loss rate and delay jitter), which can not be predicted by a Poisson model. This situation was investigated in [65] for the case of multiplexed voice packets using the *index of dispersion for intervals* (IDI), and will be summarized by the following theorem.

Theorem 1 *Let* $\{X_k,\ k \geq 1\}$ *be a sequence of interarrival times of a stationary arrival process that represents one voice source, and let* $S_k = \sum_{i=1}^{k} X_i$. *The IDI for a single traffic stream is the sequence* $\{c_k^2,\ k \geq 1\}$ *defined by*

$$c_k^2 = \frac{kVar(S_k)}{[E(S_k)]^2} = \frac{Var(S_k)}{k[E(X_1)]^2} = c_1^2 + \frac{\sum_{i,j=1,\, i\neq j}^{k} Cov(X_i,\ X_j)}{k[E(X_1)]^2},\quad k \geq 1 \qquad (2.1)$$

where $Cov(X_i,\ X_j)$ *is the covariance between* X_i *and* X_j. *For* $k > 1$, c_k^2 *measures the cumulative covariance (normalized by the square mean) among* k *consecutive interarrival times in the traffic stream. Let* c_{kn}^2 *denote the IDI associated with the superposition process of* n *independent and identically distributed arrival processes. Then,*

$$\lim_{k\to\infty} c_{kn}^2 = c_{11}^2 \gg 1 \qquad (2.2)$$

Proof: see [65]. □

Note first that large values of c_k^2, rather than individual covariances, are the cause for performance degradations; being represented in [65] by excessive packet delays

under heavy load. For a Poisson process, $c_k^2 = 1$ for all k. In other words, the superposition process of a *finite* number of voice sources deviates asymptotically from a Poisson process. A similar result applies as well to video traffic.

2.3.2 Traffic Models in ATM Networks

To capture the effect of correlations between cell arrivals in bursty streams and to accommodate the variability of arrival rates in VBR sources, a number of traffic models have been developed recently [33, 2, 52, 56, 22, 54, 46, 15]. These models have been used either as part of an analytical queueing model or to drive discrete-event simulations of buffer systems. In the following, we describe some of these models that are relevant to this research.

2.3.2.1 The MMPP Model

The MMPP is a doubly stochastic Poisson process whose arrival rate is a random variable which is modulated (i.e., controlled) by the state of a continuous-time Markov chain [14]. A 2-state MMPP was used in [22] to approximate the *aggregate* arrival rate of several voice sources. Yegani [72] analyzed an MMPP/G/1/K queueing system which implements an NTCD mechanism with one buffer and a threshold. More complicated traffic mixes can be modeled using the *superposition* principle of MMPPs: *"The superposition of two MMPP processes (with possibly different parameters) is another MMPP (with expanded state space)"* [14]. The major advantage of the MMPP model is that it captures some of the important correlations between cell arrivals, while remaining analytically tractable. However, and like any queueing approach, its computational complexity is proportional to the buffer size, which makes this model computationally impractical for systems with large buffers. The MMPP was mainly used to model voice and data traffic sources. It is still questionable whether it is an appropriate model for VBR video traffic. Modeling a heterogeneous traffic mix using

MMPP is also an open issue.

2.3.2.2 ON/OFF Models

These models capture the alternating active/idle behavior of a typical data source. A voice source with a silent detector can also be characterized using an ON/OFF model. In its basic form, an ON/OFF source alternates between burst (active) and silence (idle) periods. A burst consists of a random number of consecutive cells. For this reason, this model is often referred to as packet-train model [30]. Commonly, the number of cells in a burst is assumed to follow a geometric distribution. No cell arrivals occur during idle periods. Some analytical results for the queuing performance of a statistical multiplexer with ON/OFF traffic inputs were reported in [64].

2.3.2.3 Fluid Models

In stochastic fluid models [2, 12, 13, 33, 52, 56, 51, 66, 67, 78], a traffic source is viewed as a stream of fluid which is characterized by a flow rate. The notion of discrete arrivals of individual packets is lost as packets are assumed to be infinitesimally small. The fluid approach has been found appropriate to model the traffic in high-speed ATM-based networks for several reasons [12]. First, the burst level of the traffic in ATM networks is well-captured in fluid models. Second, the granularity of the traffic, which resulted from the standardization of small fixed-length ATM cells along with high transmission speeds (in the order of hundreds of megabits and above), makes the impact of individual cells insignificant. This provides a strong justification for the separation of time scales, which is the basis of the fluid approach. Third, the computational complexity in fluid analysis is, unlike queueing analysis, independent of the buffer size, which makes the fluid approach particularly useful for systems with large buffers.

The fluid approach was used in [12] to analyze a buffer that implements the NTCD scheme with one threshold. In [78], Zhang analyzed a two-buffer system with complete

preemptive priority for one buffer, i.e., the multiplexer dedicates up to its full capacity to the high priority buffer, with the low priority buffer being served only when the high priority buffer is empty. Other variations can be found in [67] and [51].

In its simplest from, a traffic source in the fluid approach alternates between ON (burst) and OFF (silence) periods whose durations are random with some probability distribution (often exponentially distributed). During ON periods, infinitesimally small units of information arrive at a constant rate, similar to the flow of liquid. ON periods (also OFF periods) are independent and identically distributed. For N independent voice sources (or ON/OFF sources, in general), the aggregate arrival process is modeled as an $(N+1)$-state *Markov-modulated fluid flow* (**MMFF**) process, where each state represents the number of simultaneously active sources. In addition to modeling ON/OFF sources, a fluid model was also proposed to analyze VBR video traffic from video-phones [52].

Although the fluid analysis is tractable, its numerical solution has two major computational problems: one is the possibility of 'state explosion' resulting from a large state space, and the second is the inherent *numerical instability* that results when trying to obtain solutions for finite (or partitioned) buffers. In [66] a decomposition of the state space was proposed followed by a functional transformation, which solves the first of these problems for homogeneous sources.

2.3.2.4 Time-Series Models

These models are particularly suitable for compressed video streams. The VBR output of a video compressor consists of a sequence of frames generated at a fixed rate (e.g., 30 frames/sec in the NTSC standard). Therefore, inter-frame times are constant for the whole video stream. Because of the natural visual similarities between successive images in a typical video stream, frame sizes (within a range of consecutive frames) are highly correlated. Only scene changes (and other visual discontinuities) can cause an abrupt change in the frame size. The strong correlations in the VBR

sequence, along with the constant interarrival times between successive frames, suggest that VBR traffic streams can be adequately modeled using time-series analysis and, particularly, autoregressive models.

The class of linear autoregressive models has the following general form:

$$X_n = a_0 + \sum_{r=1}^{p} a_r X_{n-r} + \epsilon_n, \quad n > 0 \tag{2.3}$$

where X_n can be the size of a frame or part of a frame such as a slice (horizontal strip in an image), and $\{\epsilon_n\}$ is a sequence of *iid* random variables, called the *residuals*, which are used to explain the randomness in the empirical data. Equation 2.3 describes an autoregressive model of order p, i.e., *AR(p)*. In [52] an *AR(1)* model was used to approximate the frame size sequence taken from a video-phone that generates a VBR stream based on a conditional replenishment compression scheme. Although this model cannot capture accurately the behavior of many empirically observed video sources, its main advantage is simplicity. Higher order autoregressive models were used in [24, 23, 61] to match the first two moments and the autocorrelation function (*acf*) of a measured VBR stream. A sophisticated *ARMA* process was proposed in [21] to model the number of ATM cells generated by a video codec (coder/decoder). With the growing complexity of video compression schemes, more complicated models, such as *Autoregressive Integrated Moving Average (ARIMA)* processes [5], are being considered. Fractal ARIMA models were first used to characterize Ethernet traffic [48]. Later, these models were found appropriate for certain types of VBR video [17]. The use of ARIMA models, in general, was motivated by empirical evidence of non-stationarity (or near-non-stationarity) in the arrival process of VBR video streams [6]. A major disadvantage of a fractal ARIMA model is the large number of parameters that are needed to describe the model (i.e., the model is non-parsimonious).

2.3.2.5 The TES Model

Correlated random variates can be generated using the Transform-Expand-Sample (TES) technique [54, 28, 29]. This technique is used to generate synthetic traffic streams to be used in trace-driven simulations. The main characteristic of this technique is that it captures both the marginal probability distribution and the autocorrelation structure of an arbitrary empirical record of data. For a given set of parameters, the *acf* of a TES-based stream has a closed-form analytical expression. Therefore, by systematically searching in the parameter space and numerically computing the resulting *acf* of the model, one is able to approximately match the *acf* of a given VBR sequence. The TES approach was used in [46] to evaluate the performance of a multiplexer whose input consists of a number of VBR video streams taken from an entertainment movie. A disadvantage of the TES model is that it requires the availability of the empirical data. It is not possible to use the parameters of the model that were obtained for a given trace to characterize other traces. Moreover, the TES model cannot capture negative correlations that are observed in MPEG streams.

2.4 The Proposed Buffer Management Scheme

2.4.1 Motivation

Current research on priority mechanisms and buffer management in ATM nodes have been mostly concerned with priorities as a preventive approach to congestion control. Only few researchers addressed the role of priorities in providing QoS guarantees to communicating sessions. The common strategy to buffer management is to provide differential service to incoming cells based on the loss or delay requirements of these cells (but not both). While this strategy could possibly provide an acceptable performance for certain traffic mixes, it is not efficient under a diverse mix of traffic which requires various loss and delay performance guarantees. In such mixes, implementing

a loss-oriented or delay-oriented priority mechanism would underutilize the available resources (buffer storage or bandwidth capacity).

To alleviate this problem, we propose a hybrid buffer management scheme that implements a *mixed* priority mechanism (i.e., based on both loss and delay priority queueing strategies). This scheme can efficiently support a diverse range of performance requirements to be expected in a multimedia high-speed networking environment. The introduction of mixed priority mechanisms, in general, can be justified as follows:

1. End-to-end delay jitter has to be reduced for services that require rapid transfer of cells.

2. Loss-sensitive cells have stringent loss requirements which can not be met by the sole implementation of a delay priority mechanism.

3. To preserve the cell sequence integrity, delay priorities should be assigned only on a call basis, whereas loss priority could be assigned on a cell basis.

4. Layer coding information may be used to simultaneously assign delay and loss priorities to cells in ATM networks.

The proposed buffer management scheme will be evaluated using sufficiently accurate traffic models. Some of these models have already been used in the literature, such as the MMPP and fluid models. New traffic models will be developed to characterize the emerging VBR video streams from MPEG coders. Video traffic will be of particular interest to this research, as it will constitute the largest proportion of the overall traffic in the future B-ISDN/ATM networks.

Once input sources have been accurately characterized, we use analysis and simulations to optimize the buffer and bandwidth resources while satisfying the QoS requirements of several types of traffic. Optimizations of buffer resources include: (1)

determining the proper number of loss and delay priority classes, (2) dimensioning of the total buffer space, and (3) partitioning of each buffer among several classes of traffic (to be accommodated within that particular buffer).

2.4.2 The General Structure of the NTCD/MB Scheme

To provide several levels of loss and delay performance guarantees for a heterogeneous mix of traffic sources arriving at an ATM node, we proposed a new buffer management scheme which is referred to as *Nested Threshold Cell Discarding with Multiple Buffers* (NTCD/MB) [40, 39, 73]. This mechanism is based on mixed loss and delay priority queueing strategies. Delay priority is supported through the use of multiple buffers (one for each class of delay priorities). Loss priority is provided on a buffer-by-buffer basis using the conventional NTCD loss priority queueing scheme. NTCD was chosen over ***push-out*** (which has a slightly better performance than NTCD [35]) because of its simplicity that makes it attractive for implementation in the fast-switching ATM environment. In NTCD/MB an ATM cell arriving at a node enters one of several buffers, depending on the delay requirements of that cell. The number of buffers is the same as the number of delay levels to be supported by the network. Cells in a given buffer are guaranteed the same amount of delay performance (with possibly some of them experiencing less delay than the guaranteed level). An NTCD mechanism can be implemented in each buffer to provide cells in that buffer with different cell loss rates, depending on the loss priorities of these cells. An example of an NTCD/MB mechanism which consists of n buffers with different number of thresholds in each buffer is shown in Figure 2.2. Note that the number of the thresholds and their values depend on the set of cell loss rates to be guaranteed by the network and the relative offered load from each class. Queues are served cyclically by a multiplexer whose capacity is divided *unequally* among the queues, favoring the ones with higher delay priority. The fractional capacities *guaranteed* for each buffer are determined

based on the actual values of delay levels to be supported by the network. The server is *work-conserving* which means that no service cycles are wasted waiting at an empty queue if another queue is not empty.

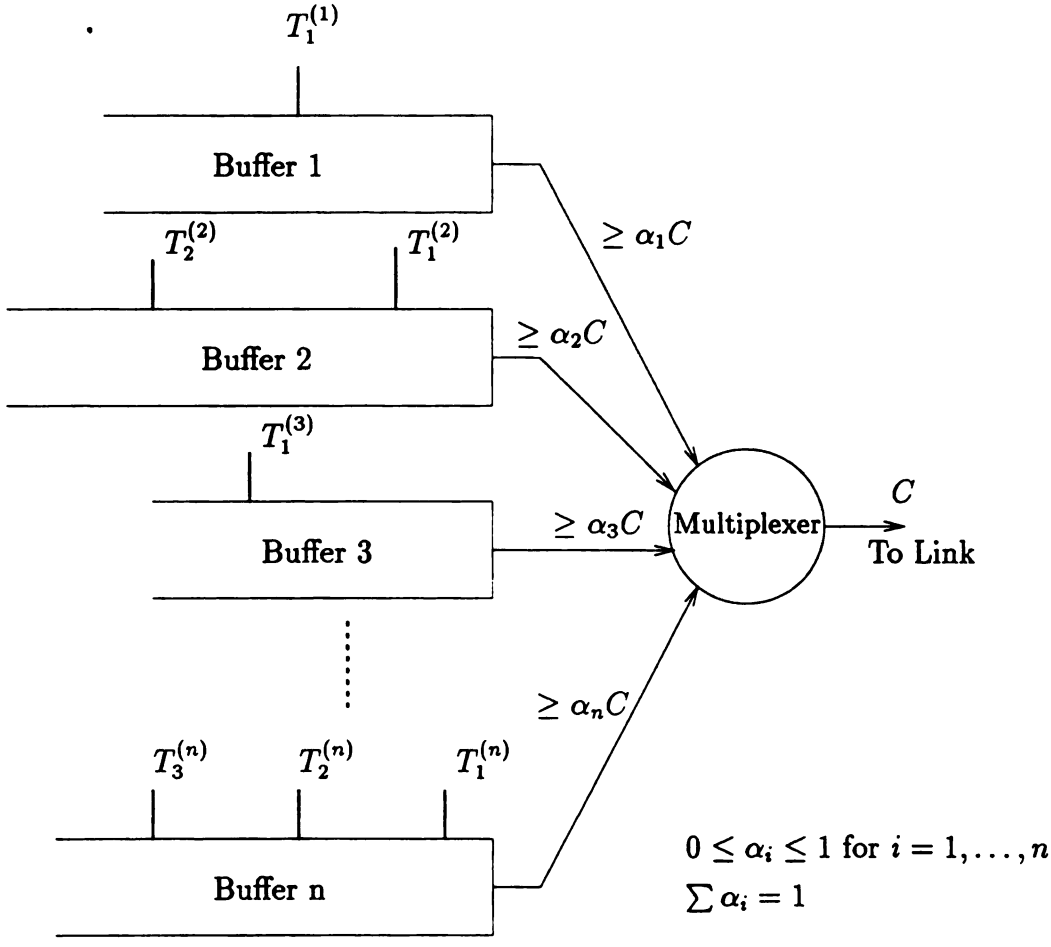

Figure 2.2: General structure of NTCD/MB.

Chapter 3

Simulation Studies of NTCD/MB

3.1 Introduction

In this chapter, we study the performance of NTCD/MB via simulations. Three case studies were used to investigate three different forms of NTCD/MB. In each case study, the input traffic consists of two general types: *real-time* (RT) traffic and *nonreal-time* (NRT) traffic. RT traffic consists of a number of voice sources while NRT traffic consists of a number of data sources. Clearly, voice traffic is more sensitive to cell delay than data traffic. Hence, two classes of *delay* priority are associated with RT and NRT traffic types. In the first case study, we additionally assume that RT traffic consists of two types of cells: loss sensitive cells, which we refer as *real high* (RH) (or class 1) and cells that are less sensitive to loss, which we refer as *real low* (RL) (or class 2). RH and RL cells are treated as two classes of *loss* priority cells. No such classes of loss priority are used in the second case study. The third study assumes that RT and NRT streams *both* contain cells of two loss requirements. The objectives of our simulations are to study the overall performance of NTCD/MB, compare this performance with that of a single-buffer NTCD, and investigate the effect of the various parameters associated with NTCD/MB.

3.2 Case Study I: Two Buffers and One Threshold

3.2.1 The Mechanism

In this case study, RT traffic (i.e., voice) consists of RH and RL cells. To support two classes of delay priority (RT and NRT), a two-buffer NTCD/MB is needed. Additionally, a threshold is required in one of the buffers to support the two classes of loss priority within RT traffic. NRT cells enter one buffer, denoted by BNR (of size B_1), while RT cells enter another buffer, denoted by BR (of size B_2). Since RT cells have two loss requirements, BR implements an NTCD mechanism with a threshold T ($0 \leq T \leq B_2$) which represents the amount of buffer space shared by RH and RL cells. If a RL cell arrives at some instant when the content of BR is above T, it is discarded. An arriving RH cell is discarded only if the content of BR was B_2. The size of BR is naturally smaller than that of BNR, and is determined by the maximum queueing delay requirements of RT cells. For the present case, the structure of NTCD/MB takes the form shown in Figure 3.1.

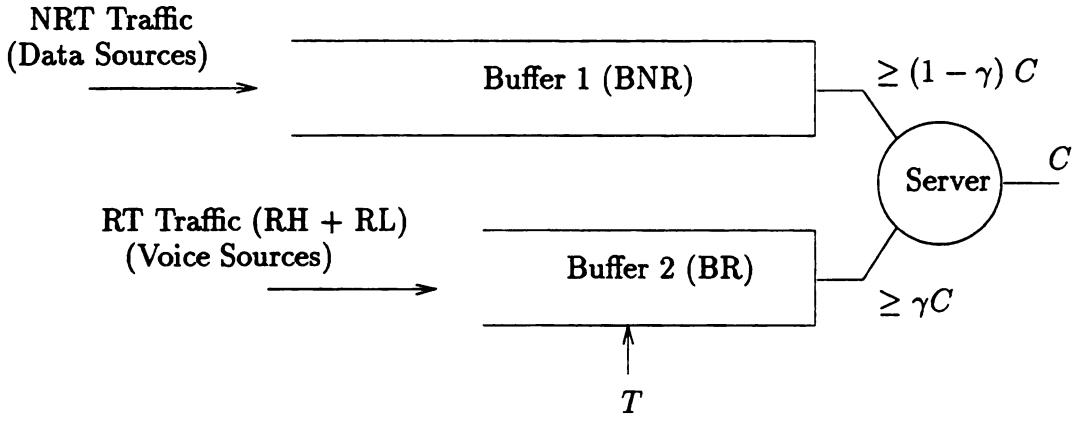

Figure 3.1: Structure of NTCD/MB mechanism.

Both buffers are served by a multiplexer whose capacity, C, is divided such that a fraction of the capacity γC (where $0 \leq \gamma \leq 1$) is guaranteed for BR, while $(1-\gamma)C$ is guaranteed for BNR. If the service discipline divides the cycles equally between the

two buffers, then the cell loss rate for RT traffic will degrade because of the small size of BR. A better approach is to serve a maximum of α RT cells followed by a maximum of β NRT cells (with $\alpha > \beta$) such that $\gamma = \frac{\alpha}{\alpha+\beta}$. The server is *work-conserving* so that if the traffic in one buffer does not completely utilize its reserved fraction of the capacity, the residual capacity is made available to the traffic in the other buffer. The selection of a specific service discipline defines the interaction between the two buffers, and can be used to facilitate the adaptation of the node to different QoS requirements.

3.2.2 Traffic Description

The packet stream from a voice source alternates between talkspurts and silence periods. During a talkspurt, cells arrive at fixed intervals. It is reasonable to assume that talkspurts and silence periods form an alternating renewal process. Nevertheless, the aggregate traffic (i.e., the superposition of a finite number of voice sources) is a complex non-renewal arrival process that possesses correlations between arrivals and fluctuations in the arrival rate. Recent studies indicated that such a traffic can be adequately characterized by a 2-state MMPP model [22, 65]. Hence, we model the aggregate RT traffic by a 2-state MMPP (denoted by MMPP_r). Similarly, the aggregate NRT traffic is modeled as a 2-state MMPP (denoted by MMPP_n). Using the MMPP model for both types of traffic is justified by the fact that voice and data streams are alike in their alternating ON/OFF behavior (of course, they differ in the distributions of the ON and OFF periods).

An MMPP, in general, is characterized by a generator matrix R and a corresponding arrival rate matrix Λ. Let the generator matrix and the arrival rate matrix for MMPP_r be

$$R_r = \begin{bmatrix} -v_{r_1} & v_{r_1} \\ v_{r_2} & -v_{r_2} \end{bmatrix}, \quad \Lambda_r = \begin{bmatrix} \lambda_r^1 & 0 \\ 0 & \lambda_r^2 \end{bmatrix} \tag{3.1}$$

where v_{r_i} is the inverse of the average sojourn time in state i for the Markov chain of MMPP$_r$, and λ_r^i is the total arrival rate when MMPP$_r$ is in state i ($i = 1, 2$). Then,

$$\lambda_r^i = \sum_k \left(\lambda_r^i\right)_k \tag{3.2}$$

where $(\lambda_r^i)_k$ is the arrival rate of class k real-time cells when MMPP$_r$ is in state i (here $k = 1, 2$). Similar expressions are used to characterize MMPP$_n$ by replacing the index r in (3.1) and (3.2) with the index n.

3.2.3 Simulation Results

Simulations of the underlying buffer system were conducted using a CSIM program[1]. The following parameters were used [40]:

- Deterministic service time which is equal to the transmission time of a cell.
- Inverse average sojourn times for the Markov chain in MMPP$_r$ are $v_{r_1} = 0.01$, $v_{r_2} = 0.03$. Likewise, $v_{n_1} = 0.04$, and $v_{n_2} = 0.01$ for MMPP$_n$. The stationary probability vector for the Markov chain of MMPP$_r$ is defined by[2]

$$\pi_r = [\pi_{r_1} \;\; \pi_{r_2}] = \left[\frac{v_{r_2}}{v_{r_1} + v_{r_2}} \;\; \frac{v_{r_1}}{v_{r_1} + v_{r_2}}\right] \tag{3.3}$$

- The real-time traffic consists of two classes of cells with arrival rates $(\lambda_r^i)_{high}$ and $(\lambda_r^j)_{low}$, where $i, j \in \{1, 2\}$, depending on the phase of the process.[2] The load ratio for MMPP$_r$ is defined as $\xi = (\lambda_r^i)_{high} / (\lambda_r^i)_{low}$, where i can be 1 or 2. This says that both classes of RT traffic are modulated by the same Markov chain (thus, ξ remains constant throughout the alternating phases of the chain).
- The server discipline is implemented with $\alpha = 2$, and $\beta = 1$.

[1] CSIM is a C-based discrete-event simulation language.

[2] Similar definition applies for MMPP$_n$ with subscript r replaced by n.

The first experiment demonstrates the performance of NTCD/MB using different values for the total offered load, ρ_{tot}, and $B_1 = 40$, $B_2 = 6$, $T = 4$, $\xi = 0.3$, and burstiness levels $\tau_r = 1.2$, $\tau_n = 2.6$ (the burstiness level is defined here as the ratio of the maximum arrival rate to the average arrival rate). Figure 3.2 shows the loss probabilities for RL and NRT cells at various ρ_{tot}. The total offered load is varied by either varying RT traffic load ρ_r and fixing NRT traffic load at $\rho_n = 0.38$, or vice versa (with $\rho_r = 0.44$). Increasing ρ_r, and therefore ρ_{tot}, results in higher cell loss rates for both types of traffic. Figure 3.3 illustrates the average response time of the system for both types of traffic. It is obvious that using NTCD/MB, RT cells of both classes experience a much smaller delay than NRT cells. From Figure 3.2 the cell loss rate of RT cells is slightly affected by increasing ρ_n. Notice that ρ_r has more impact on the performance of NRT traffic than the impact of ρ_n on RT traffic.

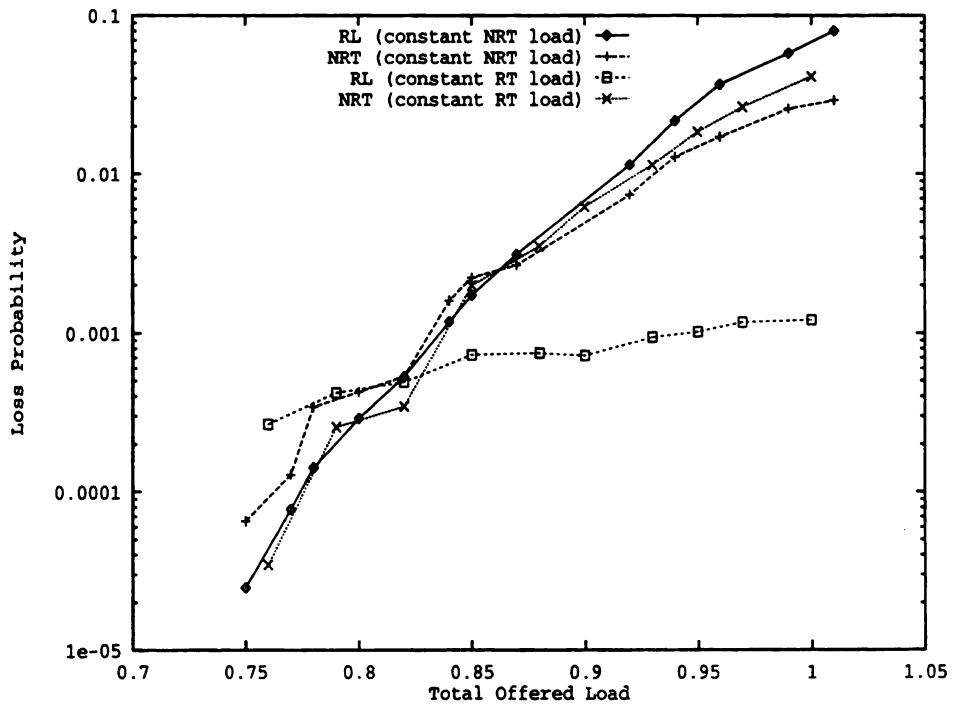

Figure 3.2: Loss probabilities versus total load using NTCD/MB.

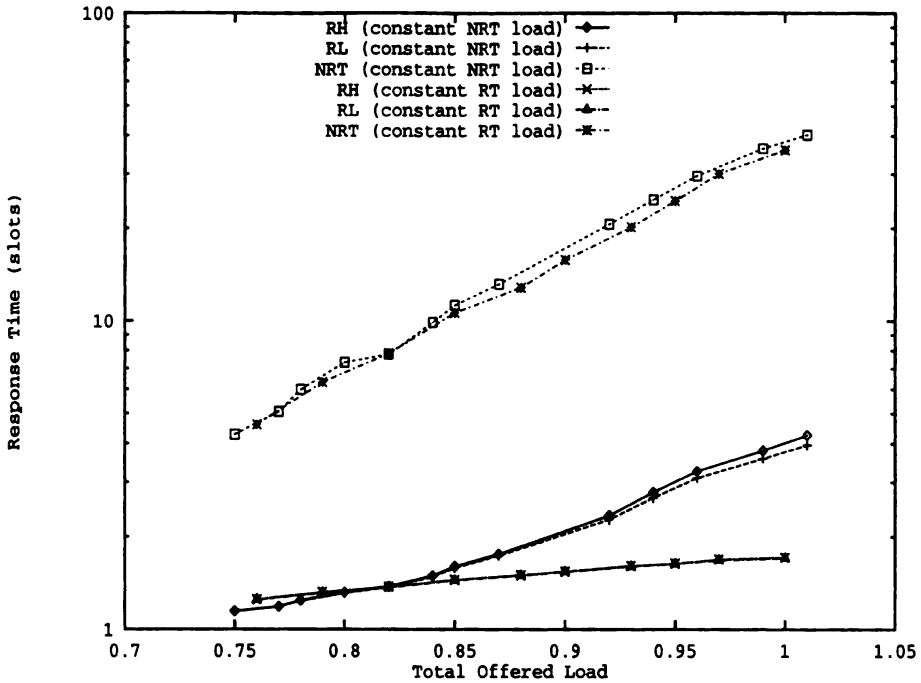

Figure 3.3: Average response time versus total offered load under NTCD/MB.

The previous simulations were repeated for a shared-buffer NTCD with buffer size $B = B_1 + B_2$. Figure 3.4 shows the average response times at various offered loads (either ρ_r or ρ_n was varied). Apparently, NTCD does not distinguish between the two types of traffic with regard to delay performance.

The loss probability of RL cells is shown in Figure 3.5 for different RT and NRT loads, using NTCD and NTCD/MB. In particular, the figure shows that in NTCD/MB the change in NRT traffic load has a very slight effect on the performance of RT cells. The effect of the threshold (T) of BR in NTCD/MB is shown in Figure 3.6 using $B_1 = 40$, $B_2 = 6$, and $\rho_{tot} = 0.82$. As expected, increasing T improves the performance for RL cells. Note that T has no effect on the loss rate of NRT cells. The response time of the system for the same experiment is shown in Figure 3.7. The choice of a particular value for T does not affect the queueing delay of cells from both types of traffic. Therefore, the value of T can be adjusted to

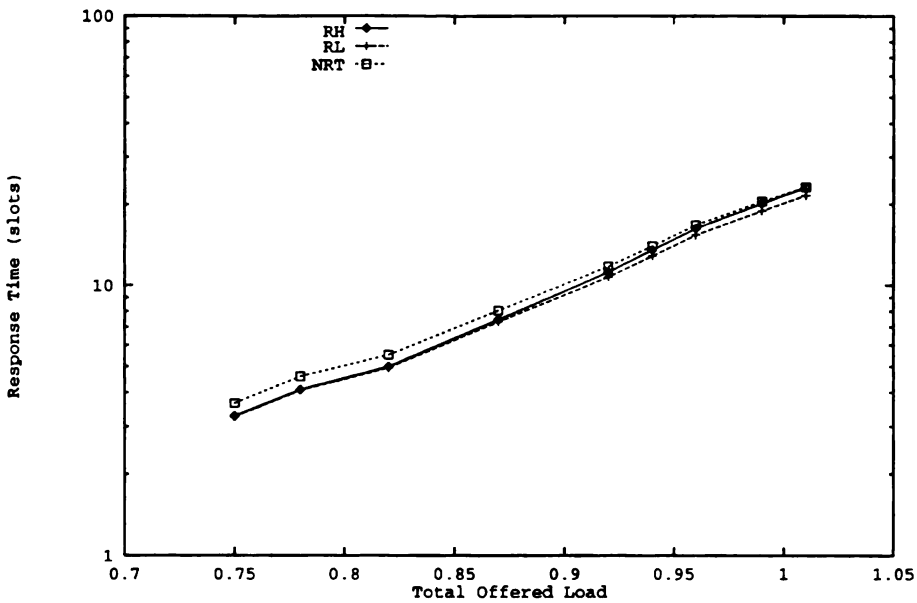

Figure 3.4: Average response time versus total offered load under NTCD.

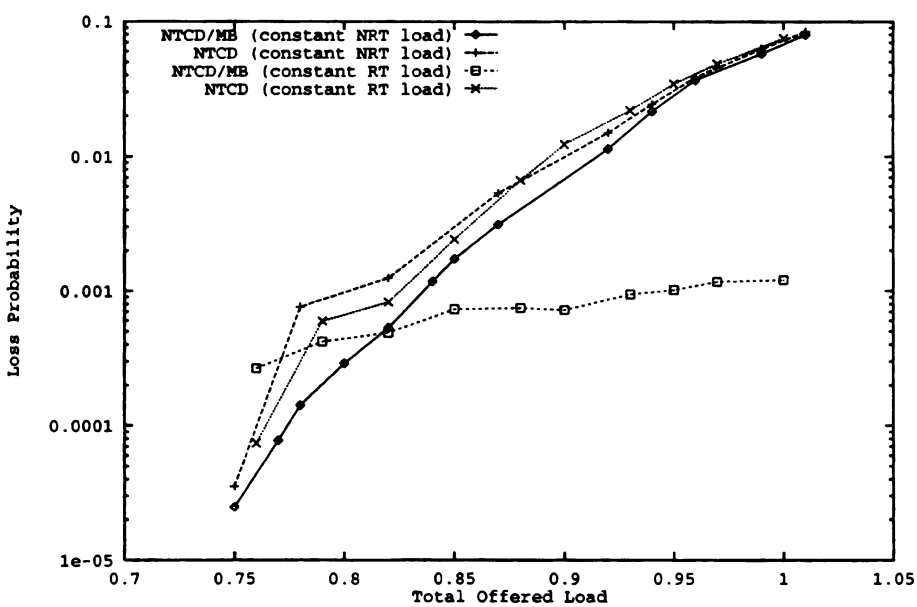

Figure 3.5: Loss probabilities versus total load for RL cells.

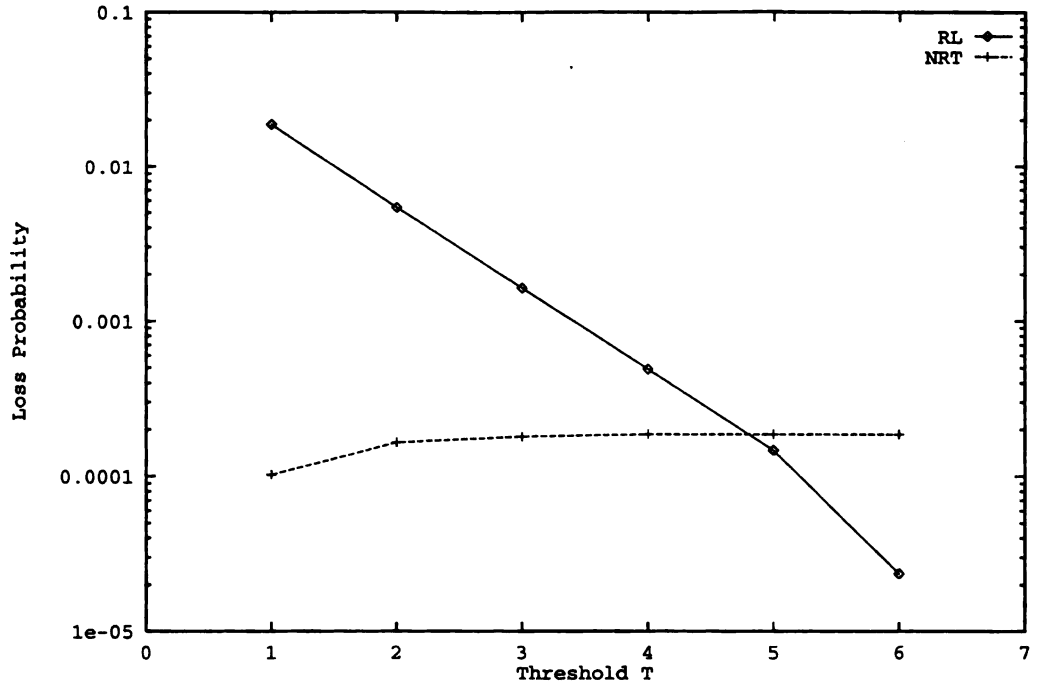

Figure 3.6: Loss probabilities versus threshold using NTCD/MB.

provide certain loss performance for RL cells independent of the delay performance experienced by these cells.

The effect of the size of BNR is shown in Figure 3.8 using $\rho_{tot} = 0.82$ ($\rho_r = 0.44$ and $\rho_n = 0.38$), $B_2 = 6$, and $T = 4$. It is clear that the loss performance for NRT traffic depends strongly on the value of B_1, whereas the loss rate for RT traffic is almost independent of B_1.

Finally, the effect of the load ratio of RT traffic on the performance is shown in Figure 3.9. This is for $\rho_{tot} = 0.9$, $\rho_r = 0.52$, $\rho_n = 0.38$, $B_1 = 40$, $B_2 = 6$, and $T = 4$. Increasing ξ tends to degrade the performance for both classes of RT cells. Obviously, the effect is more severe with RH cells because high load ratio means that larger proportion of RH cells are to be accommodated in the buffer. Therefore, more of these cells will arrive at a saturated buffer and be dropped. The loss rate of NRT cells is not affected by the load ratio of RT traffic.

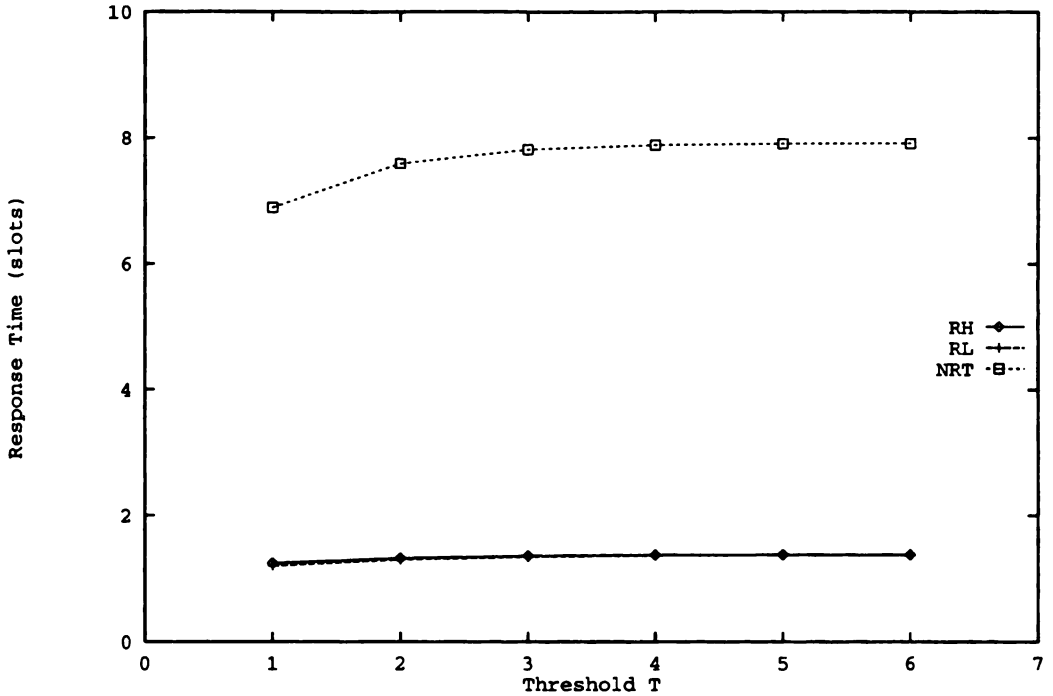

Figure 3.7: Average response time versus threshold using NTCD/MB.

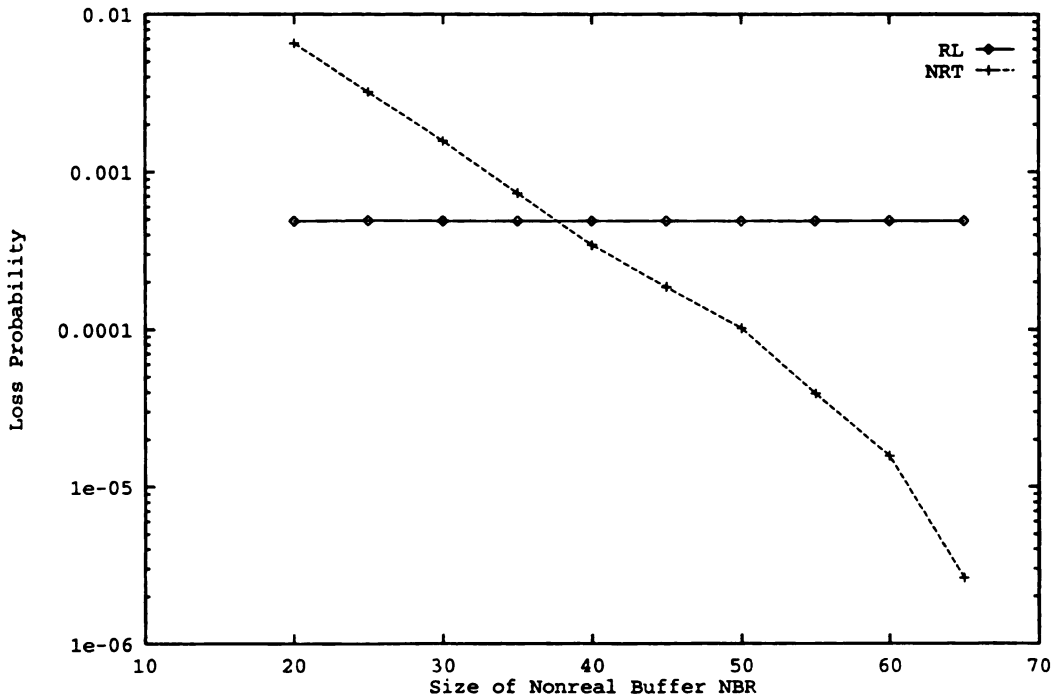

Figure 3.8: Loss probabilities versus the size of BNR using NTCD/MB.

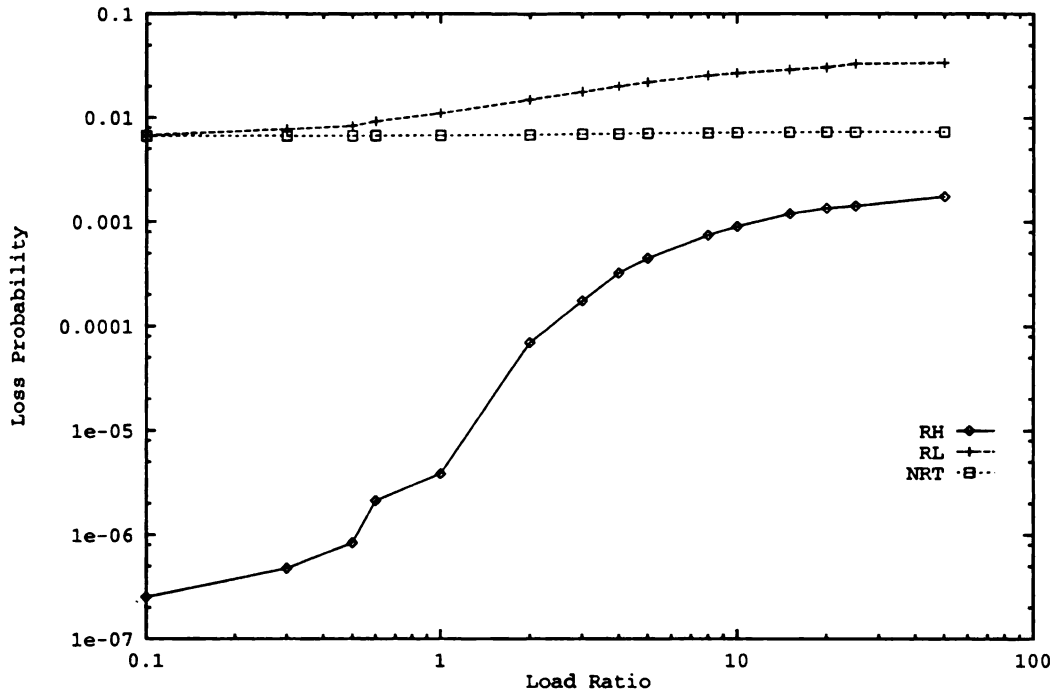

Figure 3.9: Loss probabilities versus RT load ratio using NTCD/MB.

3.3 Case Study II: Two Buffers and No Thresholds

3.3.1 The Mechanism

In the early implementations of integrated voice/data networks (i.e., ISDN), voice traffic was often given a *complete* non-preemptive priority over data [68]. Such a design can be formulated as a special case of the NTCD/MB mechanism [73]. As before, two buffers, BR and BNR, are used to accommodate RT (voice) and NRT (data) traffic types. RT does not have multiple classes of loss priority, so there is no need to implement a threshold-discarding mechanism. The structure for this particular NTCD/MB is shown in Figure 3.10.

Notice that in this case cells in BR have a *complete* scheduling priority over cells in BNR. Therefore, RT traffic can use up to the full bandwidth capacity, if needed (since a non-preemptive priority is assumed, RT traffic may experience a service rate which

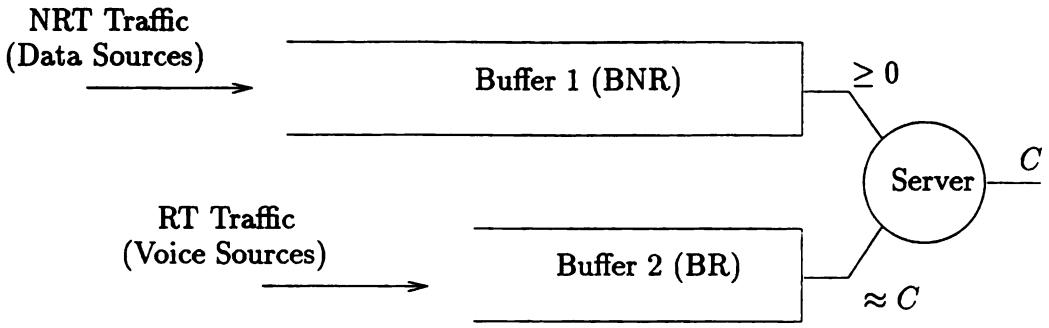

Figure 3.10: Structure of NTCD/MB in Case Study II.

is slightly smaller than C). Cells in BNR are served only when BR is empty, i.e., BNR is not guaranteed any amount of bandwidth (this is indicated by the "≥ 0" notation). Despite the popularity of such a configuration, it fails to provide any performance guarantees to data traffic (which is needed in certain cases). Nonetheless, we would like to study the performance of this configuration and compare it to that of a single-buffer NTCD (Figure 3.11). When using a single-buffer NTCD, we assume that RT traffic has a higher priority than NRT, and this priority is supported using a threshold. Notice that no distinction is being made between loss and delay priority types in the case of a single-buffer NTCD.

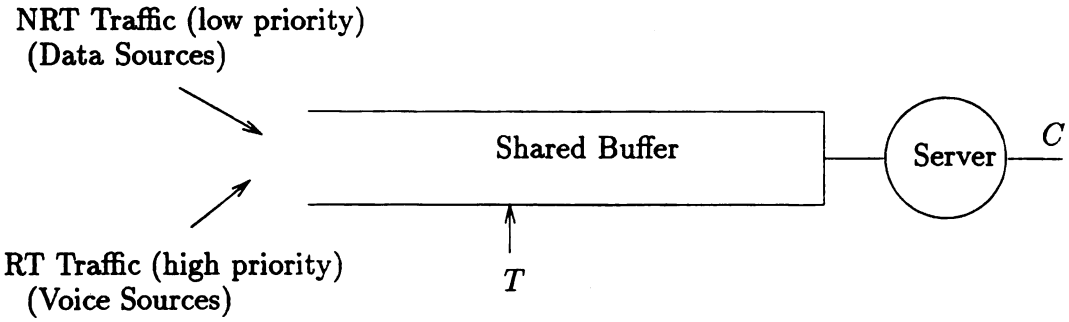

Figure 3.11: Structure of a single-buffer NTCD in Case Study II.

3.3.2 Traffic Description

As before, RT traffic consists of the superposition of several voice sources. In this case, we assume that *each* voice stream is modeled using a 2-state MMPP with average sojourn times $v_{r_1}^{-1}$ and $v_{r_2}^{-1}$ for states 1 and 2, respectively. State 1 represents the active period (i.e., talkspurt) in the stream, while state 2 represents the idle period (i.e., silence). As indicated by empirical results [22], appropriate values for the average sojourn times are: $v_{r_1}^{-1} = 352$ msec and $v_{r_2}^{-1} = 650$ msec. The average arrival rates in state 1 and 2 are, respectively, $\lambda_r^2 = 32$kbps (based on ADPCM coding) and $\lambda_r^1 = 0$. We model the aggregate NRT traffic as a Poisson process with average arrival rate λ_n. Characterizing data traffic using a Poisson process has been justified in [22]. Such an assumption is only valid when the number of data sources is very large. Therefore, their superposition tends to a Poisson process.

Notice that the model used for RT traffic in this case study is different from the one used in Case Study I in which the *aggregate* voice traffic (i.e., the superposition of several voice sources) was modeled by a 2-state MMPP. The rationale for modeling each voice stream using a 2-state MMPP in the present study is related to an important property of the MMPP, namely the superposition principle for MMPPs [14] (see Section 2.3.2.1). When each voice stream is modeled using a 2-state MMPP, the aggregate voice traffic is another MMPP with expanded state space (the number of states for this MMPP is twice the number of superposed voice streams). To avoid a state explosion situation, we let RT traffic consist of 2 voice sources only.

3.3.3 Simulation Results

In this section, we present simulation results for the NTCD and NTCD/MB mechanisms. The time unit chosen is 352 msec, which corresponds to the average duration of an ON period in a voice source. The total output capacity (C) is adjusted to attain

the desired level of utilization. Because of the complete priority given to BR in the examined NTCD/MB, BNR should always be designed with a larger size than BR. In all experiments, the size of BR is 30 cells and the size of BNR is 50 cells. Let ρ_r and ρ_n denote the RT and NRT offered loads, respectively. When using a single-buffer NTCD, it is assumed that RT has higher priority over NRT traffic. Thus, NRT cells can be admitted as long as the content of the shared buffer is below the threshold level T. Three values for T were used: 25%, 50%, and 75% of B (the total size of the shared buffer), where $B = B_1 + B_2 = 80$ cells. It is shown that the performance for both types of traffic is significantly affected by the value of T.

The performance of both mechanisms was observed in terms of the cell loss rate and average cell delay (queueing time plus service time). Simulation run times were selected so that the traffic with the smallest loss probability loses at least 100 cells. Otherwise, the results could be inaccurate. More intense simulations were run for particular experiments to confirm the correctness of the results. In all the experiments, the traffic intensities, ρ_r and ρ_n, were varied. When ρ_r is varied, ρ_n is held constant at 0.4, and vice versa.

In Figures 3.12 and 3.13, ρ_r is varied. Figure 3.12 shows the loss probability for NRT cells versus offered load based on NTCD/MB and also NTCD (with three values for T). For $\rho_{tot} \geq 0.7$ ($\rho_{tot} \triangleq \rho_r + \rho_n$), the performance of NTCD/MB lies between the performance of NTCD with $T = 40$ (in cells) and the performance of NTCD with $T = 60$. For $\rho_{tot} < 0.7$, the performance (in terms of cell loss) of NTCD/MB becomes worse than both cases. As expected, NTCD with the highest T (i.e., $T = 60$) provides the best performance for NRT traffic. The loss rate of RT cells in the same experiment is depicted in Figure 3.13. Observe that NTCD/MB has the worst relative performance (compared to NTCD using three different threshold values) under heavy load and next to best performance under light load. Also, the smaller the threshold in NTCD, the better the performance experienced by RT cells.

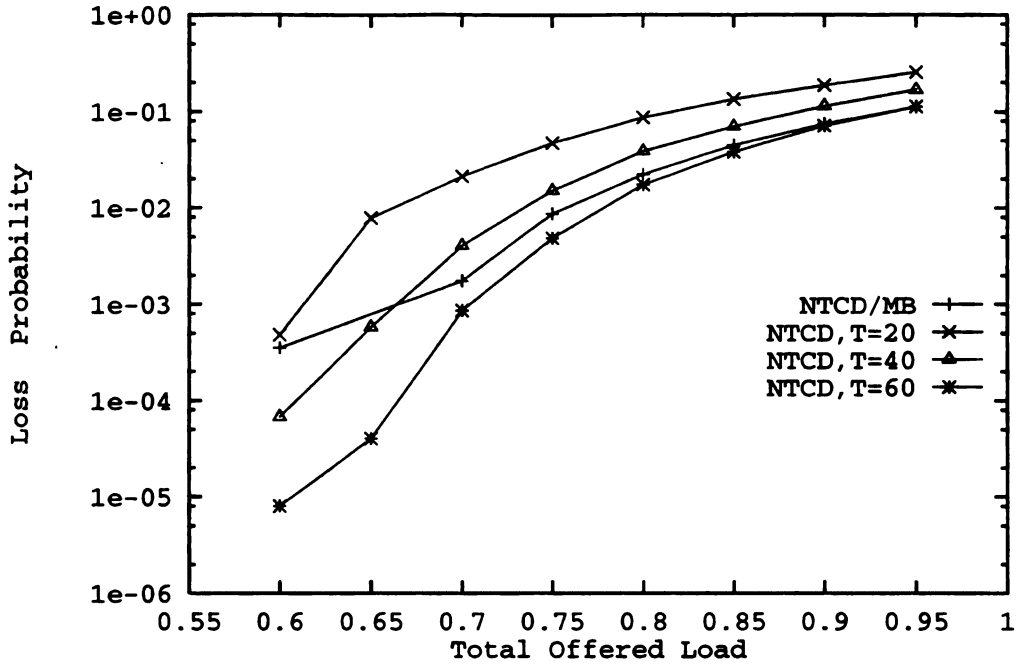

Figure 3.12: Cell loss probability for NRT traffic versus ρ_{tot} (ρ_r is varied).

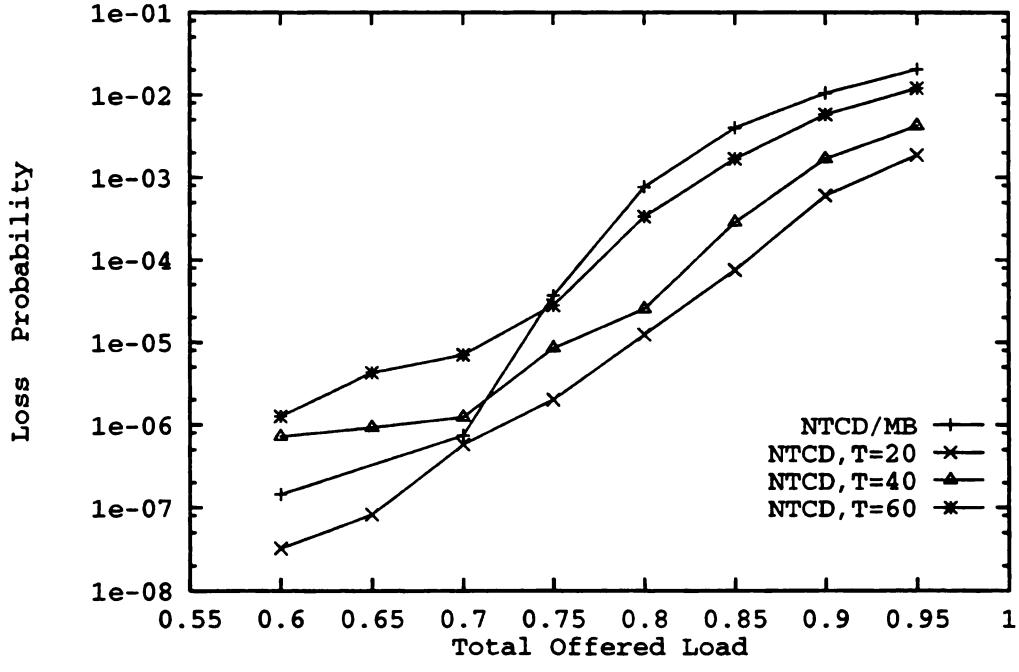

Figure 3.13: Cell loss probability for RT traffic versus ρ_{tot} (ρ_r is varied).

The same set of experiments were repeated with ρ_n varied and $\rho_r = 0.4$. Cell loss rates for RT and NRT traffic are depicted in Figures 3.14 and 3.15, respectively. In Figure 3.14, the same behavior as in Figure 3.12 is observed except for the performance of NTCD/MB under light load. Here, NTCD/MB had the best performance. It is observed that when using NTCD/MB, the cell loss rate for the RT traffic is independent of ρ_n. Although it has a worse performance than NTCD in this case, NTCD/MB can be used to dimension the buffer space by allowing the size of BR to be chosen based on the requirements of real-time traffic only.

Average delay of a cell in the previous experiments is shown in Figures 3.16 and 3.17 (with ρ_r being varied and $\rho_n = 0.4$), and in Figures 3.18 and 3.19 (with ρ_n being varied and $\rho_r = 0.4$). Figure 3.16 shows the average waiting time in the system for a NRT cell. NTCD/MB results in the longest waiting time among the four cases. This is acceptable since NRT traffic is not delay-sensitive. In contrast, the delay performance experienced by RT cells (Figure 3.17) is the best when using NTCD/MB. In this case, delay is a significant performance measure. Note that for NTCD the larger the value of the threshold the longer is the waiting time. Similar conclusions can be drawn from Figures 3.18 and 3.19 when ρ_n is varied and $\rho_r = 0.4$.

3.4 Case Study III: Two Buffers and Two Thresholds

3.4.1 The Mechanism

It is possible that both voice and data sources contain cells with two loss requirements. In this case, NRT would consist of *nonreal high* (**NH**) cells and *nonreal low* (**NL**) cells. RT traffic consists, as in Case Study I, of RL and RH cells. To support the requirements of such a traffic mix, an NTCD/MB with two buffers is needed.

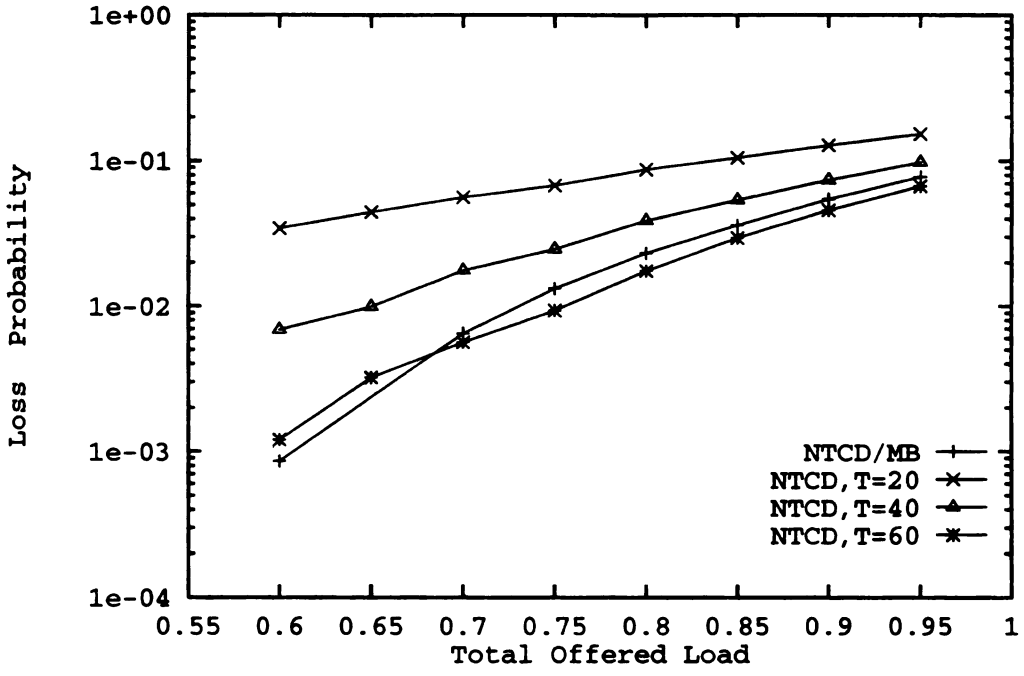

Figure 3.14: Cell loss probability for NRT traffic versus ρ_{tot} (ρ_n is varied).

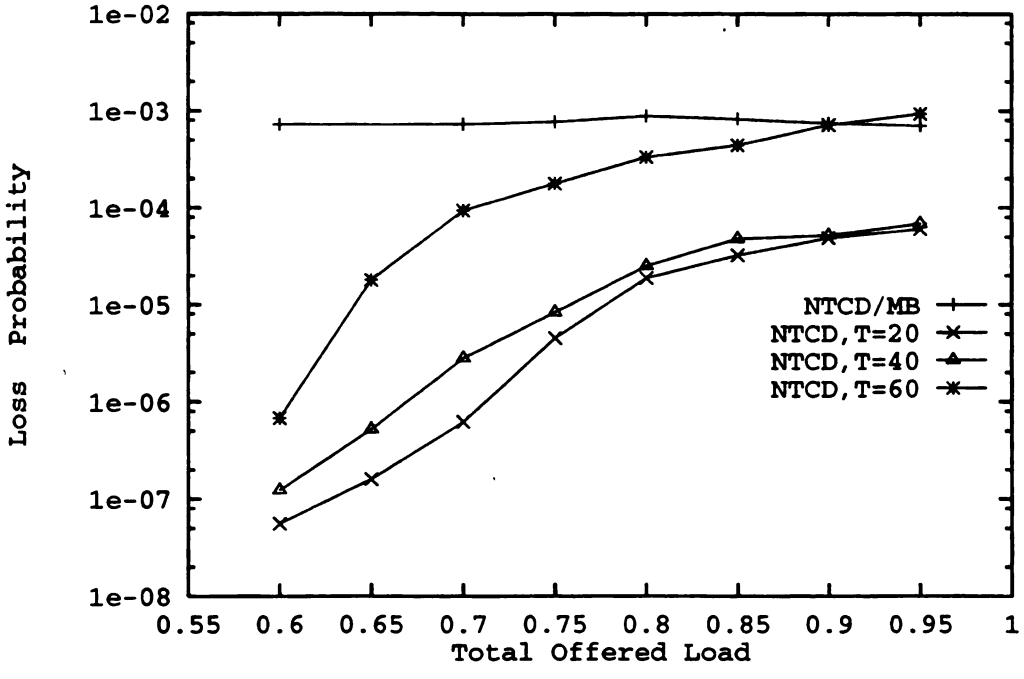

Figure 3.15: Cell loss probability for RT traffic versus ρ_{tot} (ρ_n is varied).

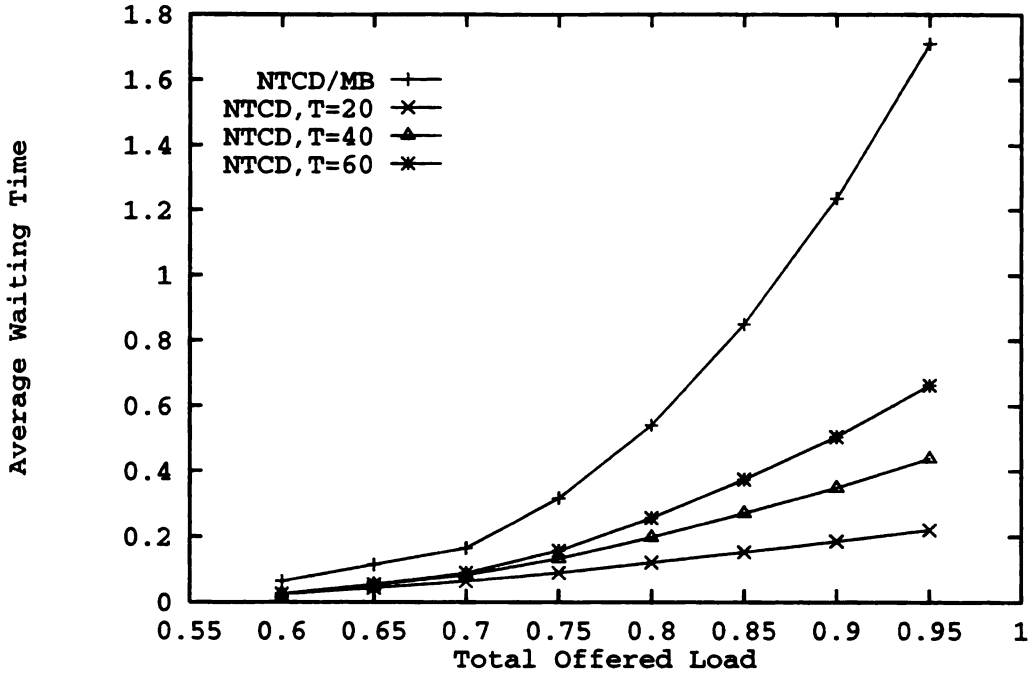

Figure 3.16: Average waiting time of a NRT cell versus ρ_{tot} (ρ_r is varied).

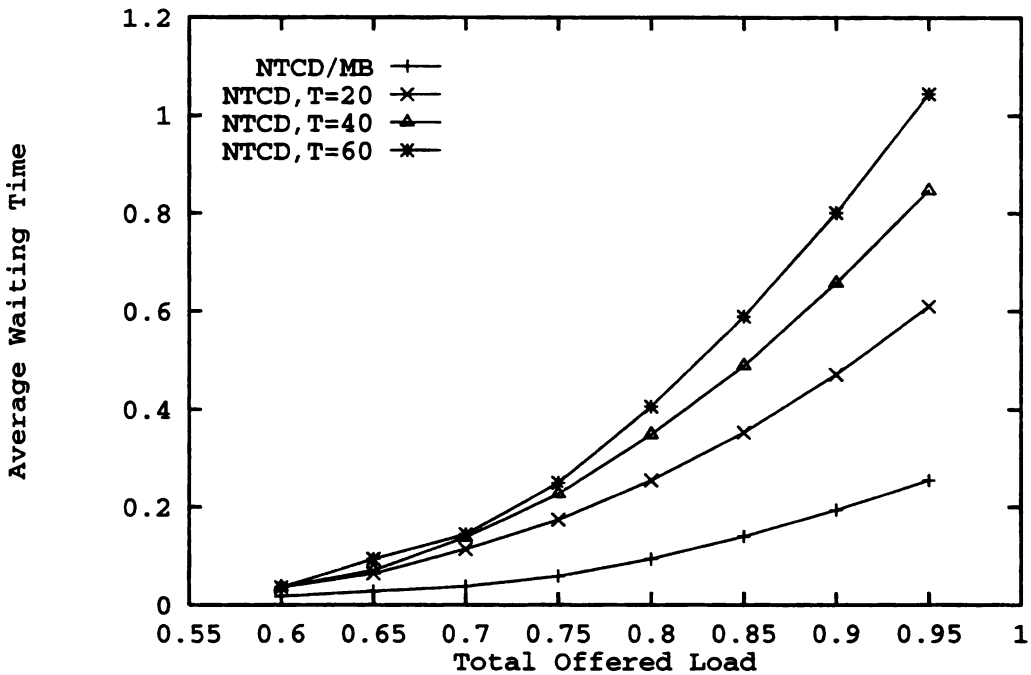

Figure 3.17: Average waiting time of a RT cell versus ρ_{tot} (ρ_r is varied).

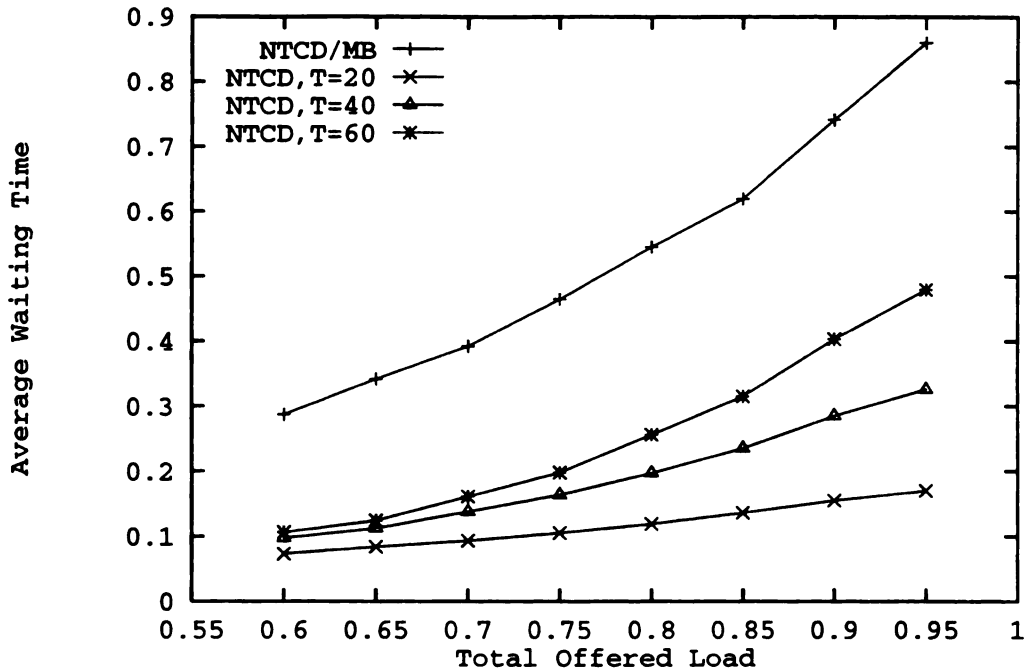

Figure 3.18: Average waiting time of a NRT cell versus ρ_{tot} (ρ_n is varied).

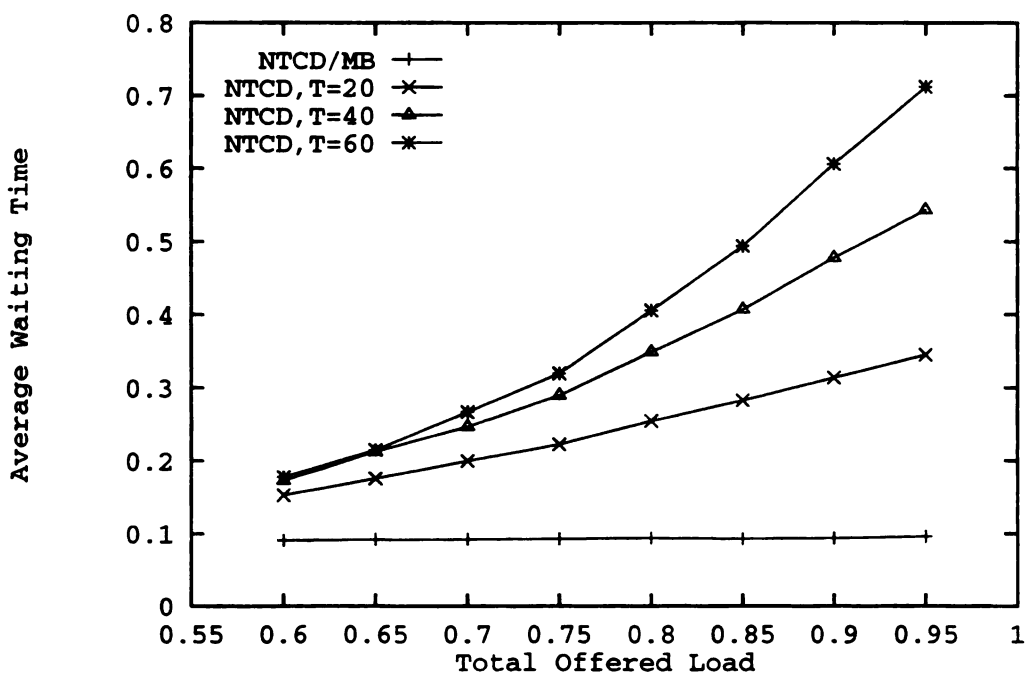

Figure 3.19: Average waiting time of a RT cell versus ρ_{tot} (ρ_n is varied).

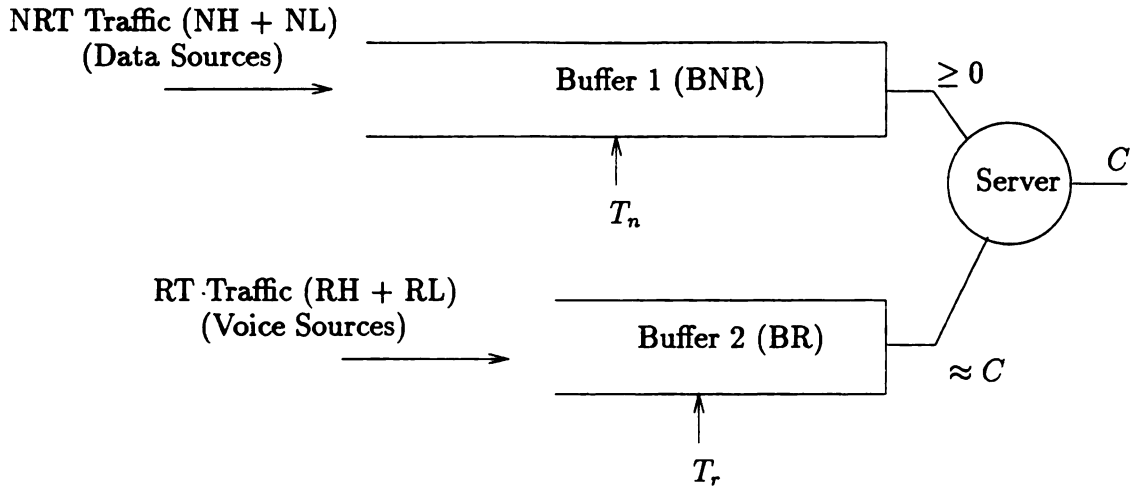

Figure 3.20: Structure of NTCD/MB in Case Study III.

Each buffer implements an NTCD with one threshold, as shown in Figure 3.20. The thresholds are denoted by T_n and T_r for BNR and BR, respectively. As in Case Study II, the scheduling discipline of the server provides complete priority to BR.

3.4.2 Traffic Description

The same traffic models in Case Study II were used to characterize RT and NRT traffic types (i.e., each voice source is a 2-state MMPP and the aggregate data traffic is a Poisson process). Let ξ_r and ξ_n be the load ratios for RT and NRT traffic, respectively. For each type of traffic, the load ratio is defined by the fraction of high priority cells among all cells of that type. We assume that RT traffic consists of two voice sources (parameters of the MMPPs are given in the previous case study).

3.4.3 Simulation Results

We investigated the effect of the offered load of both RT and NRT traffic types, the threshold of BR (T_r), and the load ratio for RT traffic (ξ_r). As before, the total offered load ρ_{tot} is varied by varying ρ_r and keeping ρ_n constant, and vice versa. In all the

results below, the sizes of the buffers (in cells) were kept at $B_1 = 40$ and $B_2 = 25$. In Figure 3.21, the cell loss rates for the various classes of traffic are shown as a function of ρ_{tot} which is varied by varying ρ_r ($\rho_n = 0.4$). The other parameters are: $T_r = 20$ (in cells), $T_n = 30$, $\xi_r = 0.4$, and $\xi_n = 0.2$. The trend in each of the performance curves is expected. Clearly, increasing ρ_r will result in higher loss rates of RH and RL cells. Since BR has a complete scheduling priority over BNR, the increase in ρ_r decreases the chances that BR is empty, thus increasing the probability of overflow at BNR. Varying ρ_{tot} via ρ_n (with $\rho_r = 0.4$) shows different trends (Figure 3.22). RT cells (both RH and RL) are not affected by the NRT traffic load. This is useful in guaranteeing the performance for RT traffic. The increase in loss probability as a function of ρ_{tot} is faster for NH cells. In general, one should expect loss probability curves to increase slowly when the loss probability is relatively large (say, above 10^{-2}). This means that the performance experienced by NL cells is less sensitive to traffic variations than that of NH cells.

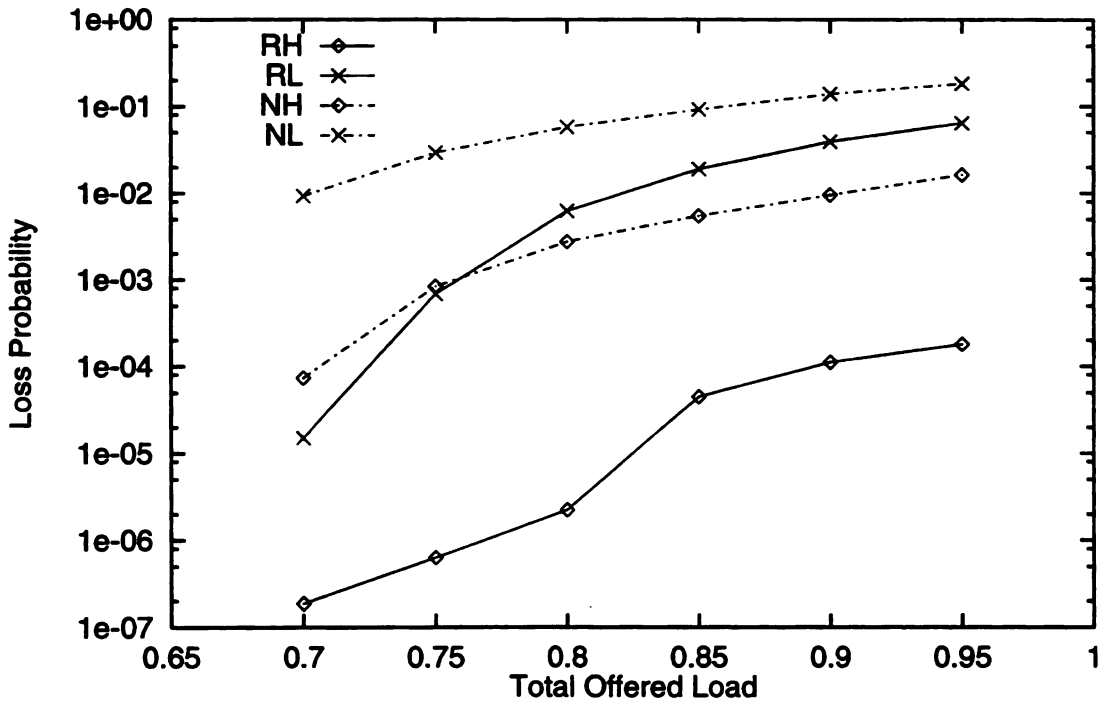

Figure 3.21: Cell loss rate versus ρ_{tot} in Case Study III (ρ_r is varied).

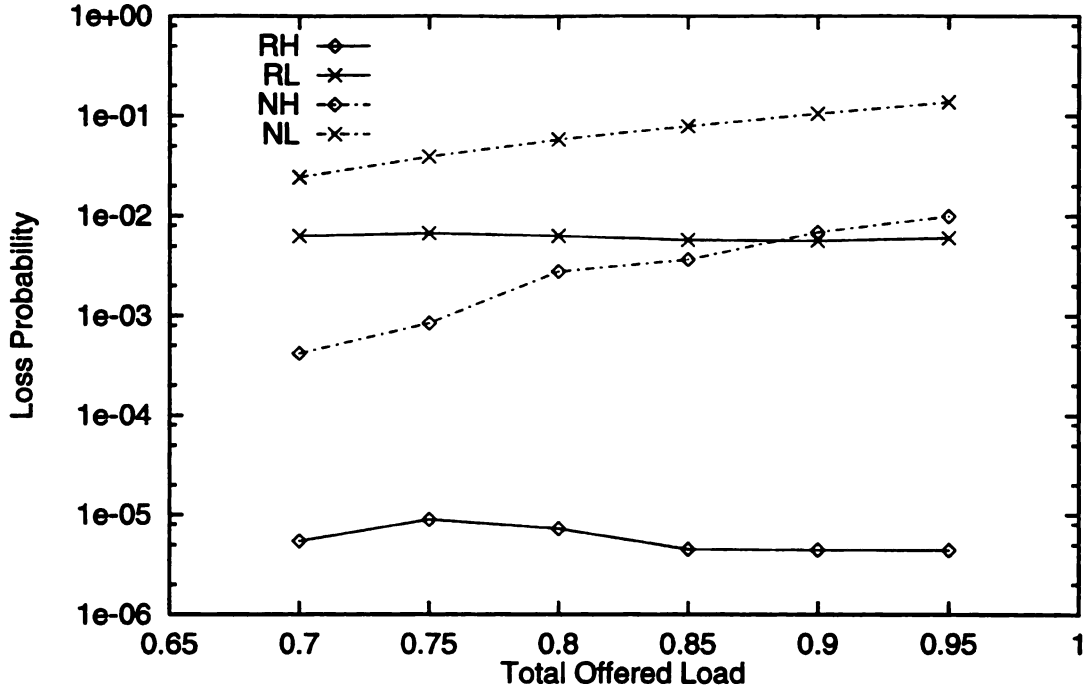

Figure 3.22: Cell loss rate versus ρ_{tot} in Case Study III (ρ_n is varied).

Figure 3.23 depicts the cell loss probability versus T_r using $T_n = 30$, $\xi_r = 0.4$, $\xi_n = 0.2$, $\rho_r = 0.6$, and $\rho_n = 0.25$ (thus, $\rho_{tot} = 0.85$). The effect of T_r on RH traffic is quite drastic. Intuitively, we expected that increasing T_r would decrease the cell loss rate of RL cells and increase that of RH cells. As shown in Figure 3.23, the loss rate of RL cells slowly decreases with T_r. This, again, can be justified by the relatively high loss rates of RL, which makes the performance for that traffic less sensitive to variations in the values of the parameters of NTCD/MB. The depicted figures are useful when dimensioning and partitioning the buffers to attain certain levels of cell loss performance. Observe that NRT traffic is slightly affected by T_r.

The effect of the load ratio of RT traffic (ξ_r) is investigated in Figure 3.24. Results shown in the figure were obtained using $T_r = 20$, $T_n = 30$, $\xi_n = 0.2$, and $\rho_{tot} = 0.85$ ($\rho_r = 0.6$ and $\rho_n = 0.25$). Increasing ξ_r causes a rapid increase in the cell loss rate for RH traffic. RL, NH, and NL traffic types are very insensitive to ξ_r.

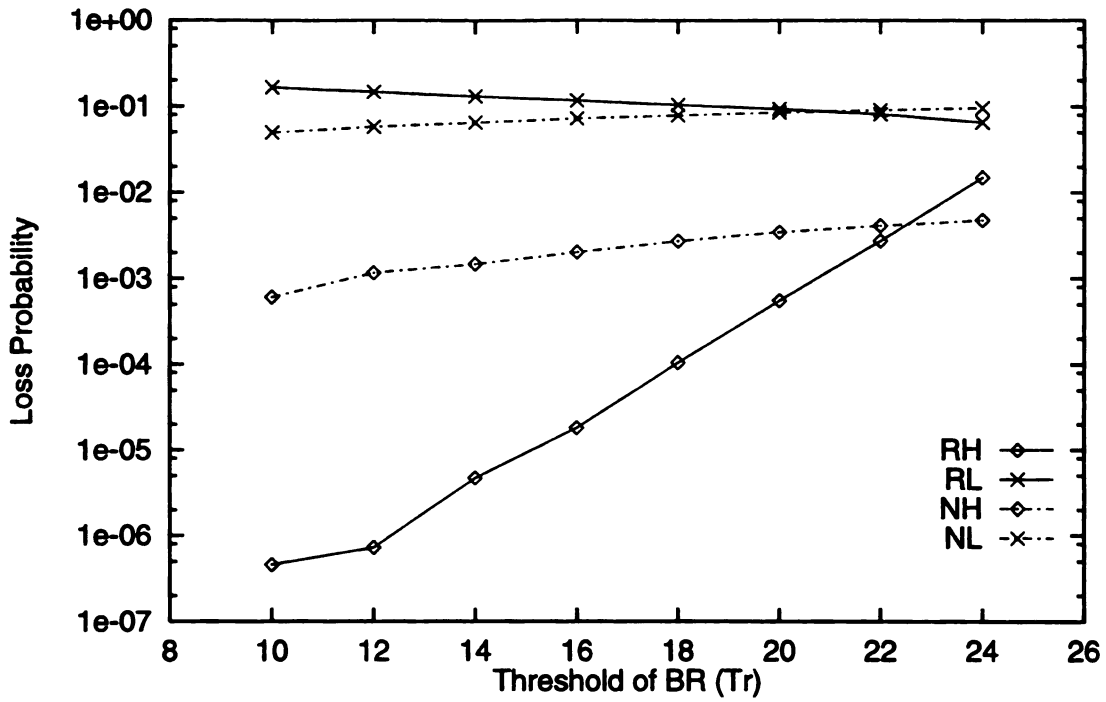

Figure 3.23: Cell loss rate versus threshold of BR in Case Study III.

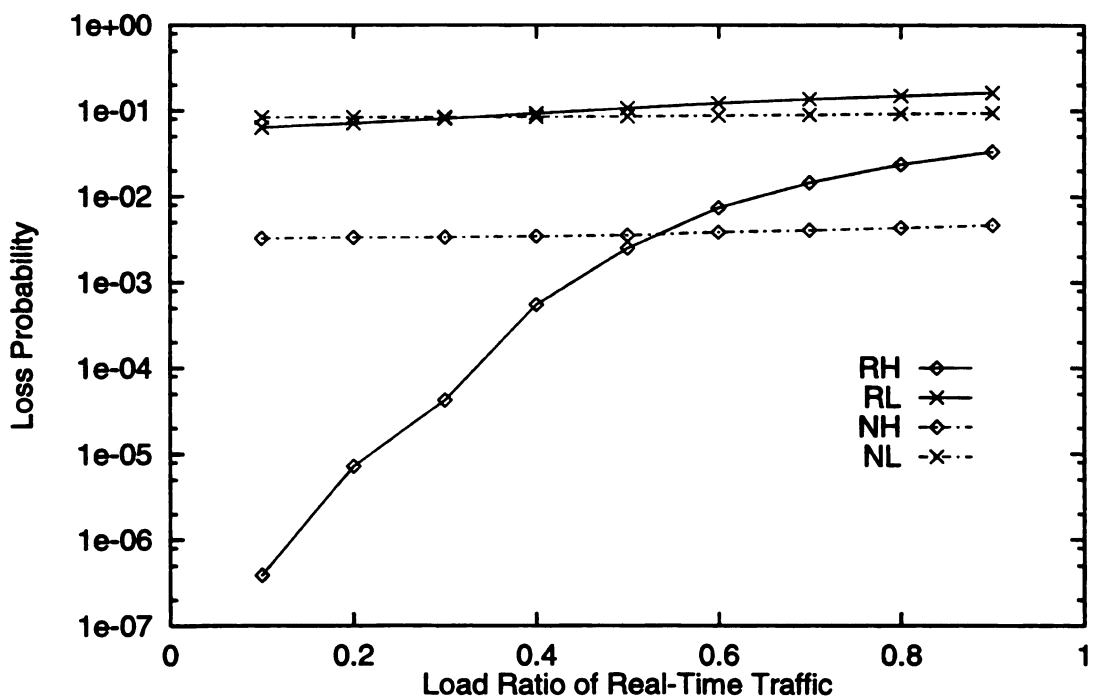

Figure 3.24: Cell loss rate versus RT load ratio in Case Study III.

Chapter 4

Fluid Analysis of NTCD/MB

4.1 Introduction

The use of simulations for performance evaluation of buffers systems becomes computationally prohibitive when the desired loss rates are below 10^{-6}. In an ATM network, some traffic streams require an end-to-end cell loss performance in the order of 10^{-10}. In such cases, performance evaluation must rely on analytical techniques to predict the performance. Most of these techniques are based on queueing analysis in which the arrival process of a traffic stream is modeled as a point process with certain stochastic characteristics.

A different approach for traffic modeling, which is adopted in this chapter, is based on stochastic fluid models [2, 33, 56]. In this approach, a traffic source ,say an ON/OFF source, is viewed as a fluid stream with random arrival rate (the flow rate). The notion of discrete arrivals of individual packets is lost as packets are assumed to be infinitesimally small. Consider, for example, the ON/OFF traffic stream depicted in Figure 4.1. Such a stream represents the behavior of a data or a voice source (with silent detectors) transmitted over a line of constant bit rate. The fluid representation for this stream is shown in part (c) of Figure 4.1.

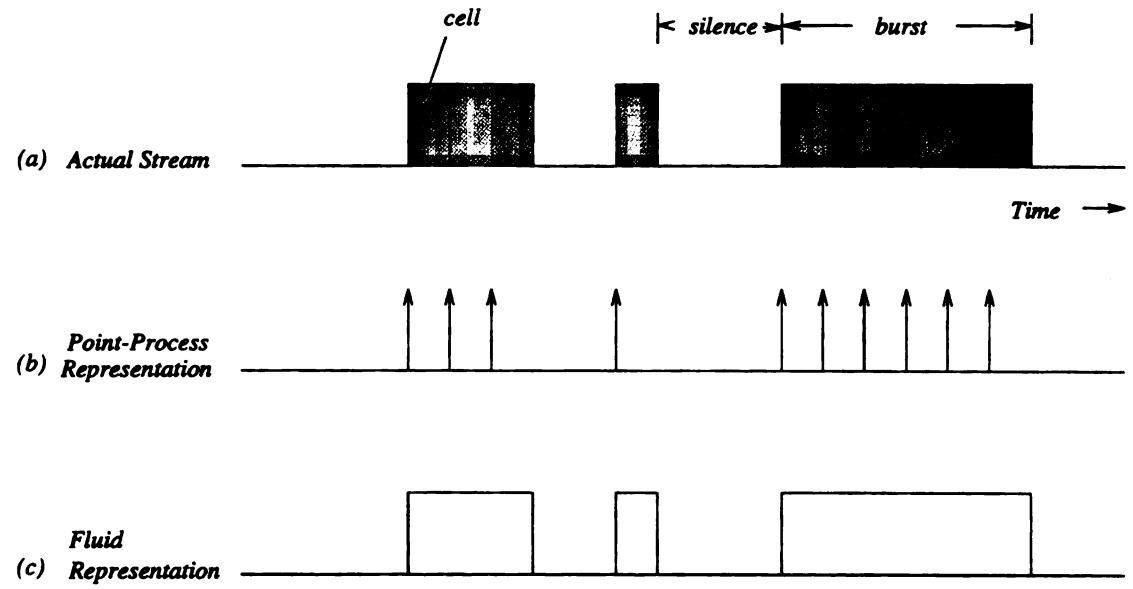

Figure 4.1: Point process and fluid representations of an ON/OFF traffic source.

The fluid approach has been found particularly appropriate to model traffic streams in high-speed ATM-based networks for a number of reasons [12]. First, this approach captures the bursty nature of traffic sources in ATM networks. Second, the standardization of small fixed-size ATM cells along with high transmission speeds (in the order of hundreds of megabits and above) makes the impact of individual cell arrivals insignificant. This provides a strong justification for the separation of time scales which is the basis of the fluid approach. Third, the computational complexity involved in fluid analysis is, unlike queueing analysis, independent of the buffer size.

In the following sections, we provide a complete fluid analysis for one particular form of NTCD/MB that was used in Case Study I of *Chapter 3*. This form is shown again in Figure 4.2. We use fluid analysis to derive the probability distributions for the content of each buffer, the cell loss probabilities, throughputs for each type of traffic, and approximate waiting time distributions [36, 41]. The results can be extended to any other form of NTCD/MB in a straightforward manner. We chose to analyze the scheme in Figure 4.2 because it contains all the main components of a general NTCD/MB: multiple buffers, thresholds, fractional guaranteed service rates,

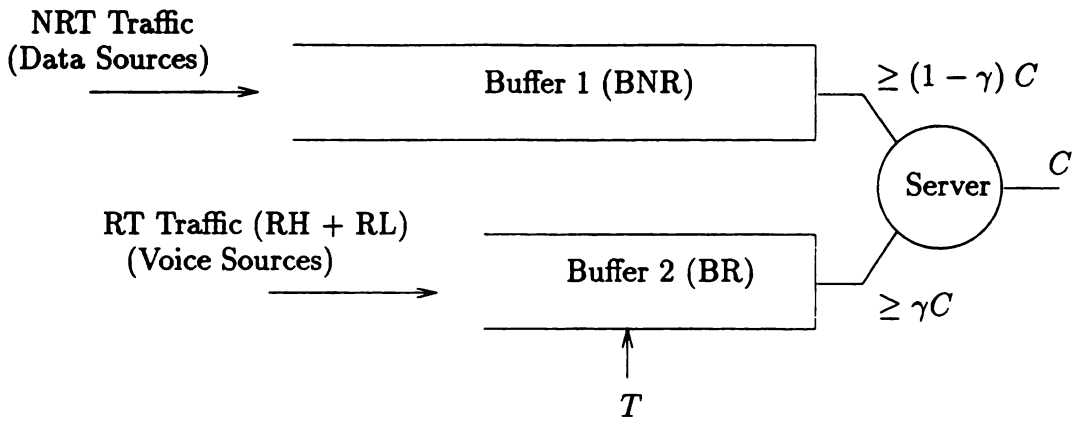

Figure 4.2: Structure of NTCD/MB mechanism.

and a work-conserving service discipline.

Elwalid and Mitra [12] analyzed a single-buffer NTCD scheme with one threshold. In [78], Zhang analyzed a two-buffer system in which complete preemptive priority is given to one buffer, i.e., the multiplexer dedicates up to its full capacity to the high priority buffer, with the low priority buffer served only when the high priority buffer is empty (this scheme is shown in Figure 3.10). *The particular significance of our analysis is that it does not assume a preemptive priority to any of the buffers analyzed; it guarantees a fractional capacity to each buffer, which can be used to provide performance guarantees; and it assumes a work-conserving service discipline.* Such extensions are by no means trivial, as they imply dependency between the contents of the two buffers. Our approach is based on obtaining the conditional probability distribution for each buffer given the content of the other buffer, then solving for the unconditional probability distributions.

4.2 Traffic Description

As in Case Study I of *Chapter 3*, we assume a traffic mix which consists of two general types of traffic: RT and NRT where, as before, RT traffic consists of two classes of

priority cells: RH and RL. The analysis to be presented is valid for a general traffic source whose arrival process can be modeled by a *Markov modulated fluid flow* (MMFF) process. NRT traffic represents N_d number of data sources, while RT traffic represents N_{vo} voice sources. Video traffic can be incorporated in the analysis using the fluid model for video traffic which was proposed by Maglaris *et al.* [52] (we will discuss this issue further in Section 4.8). However, such a model is applicable *only* to a video stream from a video-phone, using the conditional replenishment compression algorithm. It is very unlikely that this model can capture the dynamics of compressed streams from video movies, or even the dynamics of a video-phone stream which is compressed based on a compression algorithm other than the conditional replenishment technique. Characterization of video traffic is investigated in *Chapter 5* of this thesis.

Each data or voice source is characterized using the fluid representation of an ON/OFF stream. ON and OFF periods are exponentially distributed with respective average durations μ_d^{-1} and r_d^{-1} for a data source, and μ_{vo}^{-1} and r_{vo}^{-1} for a voice source. During ON periods, cells arrive at a constant rate of α_d for a data source, and α_{vo} for a voice source. Each voice source contains two classes of traffic: RH and RL. Except for a multiplying constant (i.e., the load ratio), the instantaneous arrival rates for the two classes are the same (i.e., the two classes are modulated using the same Markov chain). For convenience, we will frequently use the subscripts 1 and 2 to refer to quantities associated with BNR and BR, respectively.

Let $S_d(t)$ denote the state of the arrival process (i.e., number of data sources that are simultaneously ON) at BNR at time t. Then, $S_d(t) \in \{0, 1, \ldots, N_d\}$. The total arrival rate of NRT traffic when $S_d(t) = i$ is $\lambda_d(i) = i\alpha_d$. A transition from state i to state $i+1$ occurs when one idle data source goes ON. Since data sources are independent, and ON and OFF periods are exponentially distributed, such a transition will occur after a random time which has an exponential distribution with

rate $(N_d - i)r_d$. In a similar fashion, a transition from state i to state $i-1$ will occur after an exponentially distributed period with average rate $i\mu_d$. Accordingly, the arrival process of the aggregate NRT traffic is Markovian with a transition diagram as shown in Figure 4.3. This arrival process is an $(N_d + 1)$-state MMFF (Markov-modulated fluid flow) process [2].

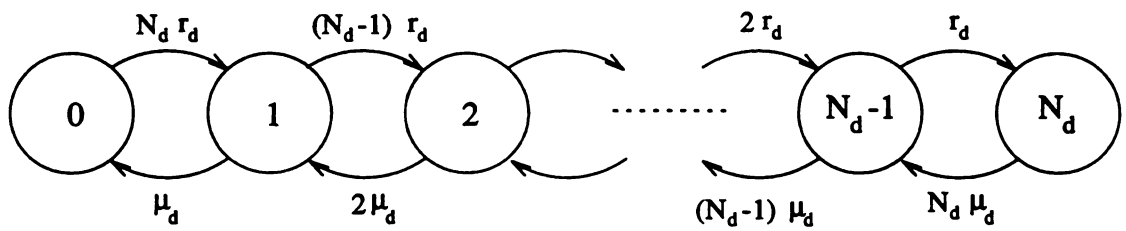

Figure 4.3: State transition diagram for a $(N_d + 1)$-state MMFF.

Let $\Pi_d = \left[\pi_0^{(d)}\ \pi_1^{(d)}\ \ldots\ \pi_{N_d}^{(d)}\right]^T$ be the stationary probability vector for the modulating chain. It is known that Π_d is a binomial distribution with parameters $(\frac{r_d}{r_d+\mu_d}, N_d)$. The j'th element of Π_d is given by

$$\pi_j^{(d)} \triangleq \lim_{t\to\infty} \Pr\{S_d(t) = j\} = \begin{pmatrix} N_d \\ j \end{pmatrix} \cdot \left(\frac{r_d}{r_d+\mu_d}\right)^j \left(\frac{\mu_d}{r_d+\mu_d}\right)^{N_d-j} \tag{4.1}$$

The effective arrival rate of NRT traffic is $\overline{\lambda}_d = \sum_j \lambda_d(j)\pi_j^{(d)}$. Let Q_d denote the generator matrix of the Markov chain. Then,

$$Q_d = \begin{bmatrix} -r_d N_d & r_d N_d & 0 & 0 & \cdots & 0 \\ \mu_d & -r_d(N_d-1)-\mu_d & r_d(N_d-1) & 0 & \cdots & 0 \\ 0 & 2\mu_d & -r_d(N_d-2)-2\mu_d & r_d(N_d-2) & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & \mu_d N_d & -\mu_d N_d \end{bmatrix}$$

A similar description applies to RT traffic. Hence, the aggregate traffic of N_{vo} voice sources is described by a $(N_{vo}+1)$-state MMFF with generator matrix Q_{vo} and

equilibrium probability vector Π_{vo} (defined similar to Π_d and Q_d). The aggregate arrival rate of RT traffic at state $S_{vo}(t) = i$ $(i = 0, 1, \ldots, N_{vo})$ is $\lambda_{vo}(i) = i\alpha_{vo}$. Let $\xi = \lambda^H_{vo}(i)/\lambda_{vo}(i)$ denote the load ratio for RT traffic ($\lambda^H_{vo}(i)$ is the arrival rate of RH cells when the RT arrival process is in state i). Note that ξ is constant and does not depend on the state of the process.

The combined total arrival rate of NRT and RT traffic can be described by a global K-state MMFF process, where $K = (N_d + 1)(N_{vo} + 1)$. At time t. the state of the Markov chain of this process is $S(t) \triangleq (S_d(t), S_{vo}(t)) \in \mathcal{L}$, where $\mathcal{L} \triangleq \{(i,j) : i = 0, 1, \ldots, N_d,\ j = 0, 1, \ldots, N_{vo}\}$. The pairs in $\mathcal{L}$ are ordered lexicographically. We say $S(t) = (i,j) = k$ where $k = 1, 2, \ldots, K$, to mean that the integer k represents the k'th ordered pair in $\mathcal{L}$. The global generator matrix $\boldsymbol{Q}$ and the stationary probability vector Π for the global arrival process are given by:

$$\boldsymbol{Q} = \boldsymbol{Q}_d \oplus \boldsymbol{Q}_{vo} \tag{4.2}$$

$$\Pi = \Pi_d \otimes \Pi_{vo} = [\pi_1\ \pi_2\ \ldots\ \pi_K]^T \tag{4.3}$$

where for every global state $k = (i,j) \in \mathcal{L}$, $\pi_k = \pi_i^{(d)}\pi_j^{(vo)}$. The symbols "$\oplus$" and "$\otimes$" above indicate the Kronecker sum and the Kronecker product, respectively.

4.3 The Conditional Equilibrium Distributions

In this section, we provide the conditional equilibrium distributions for the contents of both buffers [41]. Let $X_1(t)$ and $X_2(t)$ denote the instantaneous content of BNR and BR, respectively. In addition to the state of the arrival process, the drift rate at each buffer depends on the content of the other buffer and (for BR only) the threshold T. This dependency is caused by implementing a work-conserving service discipline. Accordingly, six cases (two for BNR and four for BR) were considered, as shown in Table 4.1. Note that the quantity $d^{(l)}_{\alpha}(k)$ in the fourth column of Table 4.1 is the drift

rate $dX_l(t)/dt$ for the case l-α at some global state $k = (i,j) \in \mathcal{L}$.

Case #	Buffer	Condition	Drift at $S(t) = k = (i,j)$
1-a	BNR	$0 < X_1 < B_1$ and $X_2 = 0$	$d_a^{(1)}(k) = \lambda_d(i) - C + \min\{\gamma C,\ \lambda_{vo}(j)\}$
1-b	BNR	$0 < X_1 < B_1$ and $X_2 > 0$	$d_b^{(1)}(k) = \lambda_d(i) - C(1-\gamma)$
2-a	BR	$0 < X_2 < T$ and $X_1 = 0$	$d_a^{(2)}(k) = \lambda_{vo}(j) - C + \min\{(1-\gamma)C$
2-b	BR	$T < X_2 < B_2$ and $X_1 = 0$	$d_b^{(2)}(k) = \lambda_{vo}^H(j) - C + \min\{(1-\gamma)C$
2-c	BR	$0 < X_2 < T$ and $X_1 > 0$	$d_c^{(2)}(k) = \lambda_{vo}(j) - \gamma C$
2-d	BR	$T < X_2 < B_2$ and $X_1 > 0$	$d_d^{(2)}(k) = \lambda_{vo}^H(j) - \gamma C$

Table 4.1: Cases used to obtain the conditional distributions.

For each of the above cases, define a drift matrix

$$\boldsymbol{D}_\alpha^{(l)} \triangleq diag\left\{d_\alpha^{(l)}(k) \ :\ k = (i,j) \in \mathcal{L}\right\} \tag{4.4}$$

where $l = 1, 2$, and $\alpha = a, b$ (if $l = 1$), or $\alpha = a, b, c, d$ (if $l = 2$). The elements along the diagonal of $\boldsymbol{D}_\alpha^{(l)}$ are ordered lexicographically. At equilibrium, the steady-state *conditional* probability distributions are defined by

$$F_\alpha^{(l)}(k,x) \triangleq \lim_{t\to\infty} \Pr\left\{S(t) = k,\ X_l(t) \le x \ /\text{cond. of case } l\text{-}\alpha\right\} \tag{4.5}$$

for all l and α. Let $\boldsymbol{F}_\alpha^{(l)}(x)$ be the column vector from $\{F_\alpha^{(l)}(k,x)\}_{k=1}^K$. Using similar development to the one in [2], the system is governed by the following set of differential equations:

$$\boldsymbol{D}_\alpha^{(l)} \frac{d\boldsymbol{F}_\alpha^{(l)}(x)}{dx} = \boldsymbol{Q}\boldsymbol{F}_\alpha^{(l)}(x), \quad \text{for all } l \text{ and } \alpha \tag{4.6}$$

For each l and α, (4.6) provides K linear differential equations. The solution for this system takes the following form:

$$\boldsymbol{F}_\alpha^{(l)}(x) = \sum_{i=0}^{K} A_\alpha^{(l)}(i)\, \boldsymbol{\Psi}_\alpha^{(l)}(i)\, \exp\left\{z_\alpha^{(l)}(i)x\right\} \tag{4.7}$$

where the set of pairs $\left\{(z_\alpha^{(l)}(i),\ \boldsymbol{\Psi}_\alpha^{(l)}(i)) \ :\ i = 1, \ldots, K\right\}$ consists of the K eigenvalues-eigenvector solutions for the generalized eigenvalue problem

$$z_\alpha^{(l)}(i)\, \boldsymbol{D}_\alpha^{(l)} \boldsymbol{\Psi}_\alpha^{(l)}(i) \ = \boldsymbol{Q}\, \boldsymbol{\Psi}_\alpha^{(l)}(i),\ \forall\, i \in \mathcal{L}. \tag{4.8}$$

4.4 Boundary Conditions

To obtain the constants $\{A_\alpha^{(l)}(i)\}$, $6\,K$ boundary conditions are required. First, define the following sets of states:

$$\begin{aligned} \mathcal{U}_\alpha^{(l)} &\triangleq \left\{m \in \mathcal{L} \ :\ d_\alpha^{(l)}(m) < 0\right\} \\ \mathcal{O}_\alpha^{(l)} &\triangleq \left\{m \in \mathcal{L} \ :\ d_\alpha^{(l)}(m) > 0\right\} \end{aligned}$$

Note that $\mathcal{U}_\alpha^{(l)}$ ($\mathcal{O}_\alpha^{(l)}$) represents the set of global states in $\mathcal{L}$ that cause an *underload* (*overload*) drift for the case l-α. The boundary conditions are:

- *Boundary conditions for BNR.*

$$F_\alpha^{(1)}(k,0) \ = \ 0 \quad \forall\, k \in \mathcal{O}_\alpha^{(1)} \quad \text{and } \alpha = a,\, b \tag{4.9}$$

$$F_\alpha^{(1)}(k,B_1) \ = \ \pi_m \quad \forall\, k \in \mathcal{U}_\alpha^{(1)} \quad \text{and } \alpha = a,\, b \tag{4.10}$$

which provides $2\,K$ equations that can be used to solve for $\{A_\alpha^{(1)}(i)\}$ where $\alpha = a,\ b$.

- *Boundary conditions for BR.*

$$F_\alpha^{(2)}(k,0) \ = \ 0 \quad \forall\, k \in \mathcal{O}_\alpha^{(2)} \text{ and } \alpha = a,\, c \tag{4.11}$$

$$F_\alpha^{(2)}(k,B_2) \ = \ \pi_k \quad \forall\, k \in \mathcal{U}_\alpha^{(2)} \text{ and } \alpha = b,\, d \tag{4.12}$$

$$F_a^{(2)}(k,T) \ = \ F_b^{(2)}(k,T) \quad \forall\, k \in \mathcal{U}_a^{(2)} \bigcup \mathcal{O}_b^{(2)} \tag{4.13}$$

$$F_c^{(2)}(k,T) \ = \ F_d^{(2)}(k,T) \quad \forall\, k \in \mathcal{U}_c^{(2)} \bigcup \mathcal{O}_d^{(2)} \tag{4.14}$$

Equations (4.11) - (4.14) provide $4K$ equations which are used to solve for $\{A_\alpha^{(2)}(i)\}$, where $\alpha = a, b, c, d$.

4.5 The Unconditional Distributions

In this section, we derive the (unconditional) equilibrium distribution for the content of each buffer, independent of the other buffer. Consider a countable mutually exclusive class of events $\{B_i\}$ that partitions an event A. Let $\Pr\{B_i\} > 0$ for all i. By the law of total probability, $\Pr\{A\} = \sum_i \Pr\{A/B_i\} \Pr\{B_i\}$. We apply the same concept to find the steady-state distribution of the buffer contents. Let

$$H(i,x) \triangleq \lim_{t\to\infty} \Pr\{S(t) = i,\ X_1(t) \le x\} \text{ for } 0 \le x < B_1 \tag{4.15}$$

$$G_1(i,x) \triangleq \lim_{t\to\infty} \Pr\{S(t) = i,\ X_2(t) \le x\} \text{ for } 0 \le x < T \tag{4.16}$$

$$G_2(i,x) \triangleq \lim_{t\to\infty} \Pr\{S(t) = i,\ X_2(t) \le x\} \text{ for } T < x < B_2 \tag{4.17}$$

In the following, $H(i,x)$, $G_1(i,x)$ and $G_2(i,x)$ are to be found from $F_\alpha^{(l)}(i,x)$. Consider first the nonreal-time buffer.

$$\begin{aligned}\Pr\{S(t) = i,\ X_1(t) \le x\} &= \Pr\{S(t) = i,\ X_1(t) \le x \ /\ X_2(t) = 0\} \cdot \Pr\{X_2(t) = 0\} \\ &\quad + \Pr\{S(t) = i,\ X_1(t) \le x \ /\ X_2(t) > 0\} \cdot \Pr\{X_2(t) > 0\}\end{aligned}$$

Hence,

$$H(i,x) = F_a^{(1)}(i,x) \cdot \Pr\{X_2 = 0\} + F_b^{(1)}(i,x) \cdot (1 - \Pr\{X_2 = 0\}) \tag{4.18}$$

(note that the parameter t that indicates time is dropped to reflect asymptotic quantities). But

$$\Pr\{X_2 = 0\} = \sum_j \Pr\{S = j,\ X_2 = 0\} = \sum_{j=1}^{K} G_1(j,0) \tag{4.19}$$

Thus,

$$H(i,x) = \left[F_a^{(1)}(i,x) - F_b^{(1)}(i,x)\right]\left[\sum_{j=1}^{K} G_1(j,0)\right] + F_b^{(1)}(i,x) \qquad (4.20)$$

which is true for $0 \leq x < B_1$, and for all $i \in \mathcal{L}$.

Next consider BR.

$$\begin{aligned}\Pr\{S=i,\ X_2 \leq x\} &= \Pr\{S=i,\ X_2 \leq x \ / \ X_1 = 0\} \cdot \Pr\{X_1 = 0\} \\ &\quad + \Pr\{S=i,\ X_2 \leq x \ / \ X_1 > 0\} \cdot \Pr\{X_1 > 0\}\end{aligned}$$

Thus, for $0 \leq x < T$

$$G_1(i,x) = F_a^{(2)}(i,x) \cdot \left[\sum_{j=1}^{K} H(j,0)\right] + F_c^{(2)}(i,x) \cdot \left[1 - \sum_{j=1}^{K} H(j,0)\right] \qquad (4.21)$$

Similarly, for $T < x < B_2$,

$$G_2(i,x) = F_b^{(2)}(i,x) \cdot \left[\sum_{j=1}^{K} H(j,0)\right] + F_d^{(2)}(i,x) \cdot \left[1 - \sum_{j=1}^{K} H(j,0)\right] \qquad (4.22)$$

Since Equation (4.21) is valid for $0 \leq x < T$, at $x = 0$

$$G_1(i,0) = F_a^{(2)}(i,0) \cdot \left[\sum_{j=1}^{K} H(j,0)\right] + F_c^{(2)}(i,0) \cdot \left[1 - \sum_{j=1}^{K} H(j,0)\right] \qquad (4.23)$$

Substituting $G_1(i,0)$ from (4.23) in (4.20) and evaluating at $x = 0$, we have

$$\begin{aligned}H(i,0) &= F_b^{(1)}(i,0) + \left[F_a^{(1)}(i,0) - F_b^{(1)}(i,0)\right] \\ &\times \left[\sum_{j=1}^{K}\left\{F_a^{(2)}(j,0) \cdot \left[\sum_{p=1}^{K} H(p,0)\right] + F_c^{(2)}(j,0)\left[1 - \sum_{p=1}^{K} H(p,0)\right]\right\}\right] \qquad (4.24)\end{aligned}$$

This is a set of K linear equation of K unknowns; $H(i,0)$, $i \in \mathcal{L}$, and can be easily solved. With $H(i,0)$ evaluated, $G_1(i,0)$ can be obtained from (4.23), and thereafter,

$H(i,x)$ from (4.20). Also, $G_1(i,x)$ and $G_2(i,x)$ can be found from (4.21) and (4.22), which completes the solution.

4.6 Performance Statistics

Once the steady-state distributions for the content of BNR and BR are determined, it is straightforward to obtain the throughput and loss probabilities for the various types of traffic. First, the following quantities are computed:

$$\begin{aligned}
\Pr\{S=i,\ \text{BNR is full}\} &= \pi_i - H(i,B_1) \\
\Pr\{S=i,\ \text{BNR is empty}\} &= H(i,0) \\
\Pr\{S=i,\ \text{BR is full}\} &= \pi_i - G_2(i,B_2) \\
\Pr\{S=i,\ \text{BR is empty}\} &= G_1(i,0) \\
\Pr\{S=i,\ X_2=T\} &= G_2(i,T) - G_1(i,T)
\end{aligned}$$

The throughput for a given class of cells is the difference between the generation rate for that class and its loss rate. For NRT cells, the generation rate is $\sum_{i=1}^{K} \lambda_1(i)\pi_i$, where $\lambda_1(i)$ is the arrival rate of NRT traffic when $S=i$ (since i here is a global state, $\lambda_1(i) = \lambda_d(i \text{ modulo } N_{vo})$). Loss rate for NRT cells is the positive drift rate into BNR when BNR is full. Therefore,

$$\text{loss rate for nonreal cells } = \sum_{i=1}^{K} [\lambda_1(i) - \max\{(1-\gamma)C,\ C-\lambda_2(i)\}] \cdot (\pi_i - H(i,B_1))$$

where $\lambda_2(i)$ is the arrival rate of RT traffic when $S=i$. (Again, this is the same as $\lambda_{vo}(j)$ where j is the second element in the ith tuple). Let T_1 denote the throughput for NRT cells, T_2^H and T_2^L denote the throughput for high and low priority RT cells, respectively. (Note that we will continue to use the letter T to denote the threshold

BR). Then,

$$\begin{aligned} T_1 &= \sum_{i=1}^{K} \lambda_1(i)\pi_i - \left[\sum_{i=1}^{K} \left(\lambda_1(i) - \max\{(1-\gamma)C,\ C - \lambda_2(i)\}\right)(\pi_i - H(i,B_1))\right] \\ &= \sum_{i=1}^{K} \lambda_1(i)H(i,B_1) + \sum_{i=1}^{K} (\pi_i - H(i,B_1)) \cdot \max\{(1-\gamma)C,\ C - \lambda_2(i)\} \quad (4.25) \end{aligned}$$

$$T_2^H = \sum_{i=1}^{K} \lambda_2^H(i)\pi_i - \left[\sum_{i=1}^{K} \left(\lambda_2^H(i) - \max\{\gamma C,\ C - \lambda_1(i)\}\right)(\pi_i - G_2(i,B_2))\right] \quad (4.26)$$

where $\lambda_2^H(i) = \xi\lambda_2(i)$. Low priority real-time (RL) cells are lost whenever $X_2 > T$ in which case all generated cells are lost, or when $X_2 = T$ which results in a partial loss rate $\lambda_2(i) - \max\{\gamma C,\ C - \lambda_1(i)\}$. Therefore, the throughput for RL cells is

$$\begin{aligned} T_2^L &= \text{throughput } |_{X_2>T} + \text{throughput } |_{X_2=T} \\ &= \left\{\sum_{i=1}^{K} \lambda_2^L(i)\pi_i - \sum_{i=1}^{K} \lambda_2^L(I)\,(\pi_i - G_2(i,T))\right\} \\ &- \left\{\sum_{i\in O_a^{(2)} \bigcap U_b^{(2)}} (G_2(i,T) - G_1(i,T))\,(\lambda_2(i) - \max\{\gamma C,\ C - \lambda_1(i)\})\right\} \quad (4.27) \end{aligned}$$

In the expression above, $\pi_i - G_2(i,T)$ is the probability of $\{X_2 > T \text{ and } S = i\}$. Also, $\lambda_2^L(i) = (1-\xi)\lambda_2(i)$.

Let L_1 denote the loss probability for nonreal cells, L_2^H and L_2^L denote the loss probability for high and low priority real cells, respectively. Then,

$$L_1 = 1 - \frac{T_1}{\sum_{i=1}^{K} \pi_i\lambda_1(i)} \quad (4.28)$$

$$L_2^H = 1 - \frac{T_2^H}{\sum_{i=1}^{K} \pi_i\lambda_2^H(i)} \quad (4.29)$$

$$L_2^L = 1 - \frac{T_2^L}{\sum_{i=1}^{K} \pi_i\lambda_2^L(i)} \quad (4.30)$$

In conventional fluid analysis, the distribution for the waiting time of a cell is readily obtained from the distribution for the buffer content. However, in the model consid-

ered, this is not trivial since the service rate for a given class of cells is not constant. For NRT cells, the service rate at state $i \in \mathcal{L}$ is

$$C_1(i) = \max\{(1-\gamma)C,\ C - \lambda_2(i)\}$$

Similarly, the service rate for high and low priority real cells is

$$C_2(i) = \max\{\gamma C,\ C - \lambda_1(i)\}$$

which, for both cases, is a function of the state i. An approximate expression for the waiting time distribution of cells can be obtained using average values of C_1 and C_2 given by

$$\overline{C_1} = \sum_{i=1}^{K} C_1(i)\pi_i \quad \text{and} \quad \overline{C_2} = \sum_{i=1}^{K} C_2(i)\pi_i \tag{4.31}$$

Let W_1, W_2^H, and W_2^L denote the waiting time for a cell belonging to NRT, RH, and RL traffic types, respectively. Then,

$$\Pr\{W_1 \le t\} \approx \frac{1}{T_1}\sum_{i=1}^{K} \lambda_1(i) H(i, \overline{C_1}t) \quad \text{for } 0 \le t \le B_1/\overline{C_1} \tag{4.32}$$

$$\Pr\left\{W_1 = B_1/\overline{C_1}\right\} \approx \frac{\overline{C_1}}{T_1}\left[1 - \sum_{i=1}^{K} H(i, B_1)\right] \tag{4.33}$$

For high priority real-time cells,

$$\Pr\left\{W_2^H \le t\right\} \approx \begin{cases} \frac{1}{T_2^H}\sum_{i=1}^{K} \lambda_2^H(i) G_1(i, \overline{C_2}t) & \text{for } 0 \le t \le T/\overline{C_2} \\ \frac{1}{T_2^H}\sum_{i=1}^{K} \lambda_2^H(i)\left(G_2(i,T) - G_1(i,T)\right), & \text{for } t = T/\overline{C_2} \\ \frac{1}{T_2^H}\sum_{i=1}^{K} \lambda_2^H(i) G_2(i, \overline{C_2}t), & \text{for } T/\overline{C_2} < t < B_2/\overline{C_2} \end{cases}$$

$$\Pr\left\{W_2^H = B_2/\overline{C_2}\right\} \approx \frac{\overline{C_2}}{T_2^H}\left[1 - \sum_{i=1}^{K} G_2(i, B_2)\right] \tag{4.34}$$

Similar expressions can be obtained for low priority real-time cells.

4.7 Numerical Investigations

4.7.1 Computational Considerations

Numerical results were obtained for the analysis in the previous sections using a C program supported by Matlab library functions. Solutions for the eigenvalue/eigenvector sets were based on the decomposition technique developed in [66]. In this technique, an aggregate traffic that consists of several independent and separable rate-modulated input processes arriving at a buffer system is analyzed by decomposing the system into smaller subsystems to be treated separately. The final solution is then obtained by superposing the individual results. This approach considerably reduces the number of required computations. The other major numerical problem which we encountered is the numerical instability of the mathematical system. Such a problem is known to exist in fluid approximation of finite buffers [67]. The eigenvalues associated with such buffer systems can take positive as well as negative values. When solving for the constants $\left\{A_{\alpha}^{(l)}(i)\right\}$, the exponentiation of positive and negative terms produces a badly-scaled matrix whose values can be close to the precision of the machine. Typically, researchers have dealt with such problem by using the complementary infinite buffer distribution as an approximation to the actual finite buffer case, since the former case does not produce positive eigenvalues. In our case, however, positive eigenvalues cannot be avoided because of the finite threshold (T). We used several scaling methods to condition the matrices involved. Row as well as column scaling were required using block diagonal matrices to obtain acceptable results.

4.7.2 Numerical Results and Simulations

Numerical examples based on the previous fluid analysis were used to study the effect of various parameters on the performance of NTCD/MB. Additionally, simulation experiments were conducted and their results compared to the analytical results.

In these simulations, the traffic consists of voice and data sources with voice sources generating RH and RL cells. Each source in the simulation experiments behaves as an ON/OFF stream in which the duration of an ON period has a truncated exponential distribution to provide an integer number of time slots. A time slot corresponds to the transmission time of a cell (with respect to the input line rate), or equivalently the cell interarrival time during a burst. During an active period of a stream, cells arrive one at a time at the beginning of a slot. OFF periods in the simulation model are identical to those of the fluid model. To provide a fractional guaranteed bandwidth (equivalent to γC, and $(1-\gamma)C$ in the analysis), a cyclic service discipline is implemented in the simulation model in which the server alternates between the two buffers forming cycles of service. Each cycle consists of a maximum of α slots for BR followed by a maximum of β slots for BNR such that $\gamma = \frac{\alpha}{\alpha+\beta}$ with smallest possible integer-valued α and β (e.g., if $\gamma = 0.4$, then $\alpha = 2$, $\beta = 3$). The service discipline is work-conserving. Buffers used in the simulations are of a finite size, whereas the numerical results for the analysis were mostly obtained assuming infinite buffers to avoid numerical instability and machine precision overflow (note, however, that BR still uses a finite threshold).

It should be noted that because of the several differences between the simulation setup and the assumptions in the analytical approach (namely, the use of finite versus infinite buffers, the truncation of ON periods in the simulations, the separation of time scales in the fluid approach, and the server arbitration), the results obtained from the two approaches are not expected to be identical. However, for most of the cases both approaches gave sufficiently close results, and in all cases they indicated the same performance trends.

The parameters used in the following experiments are normalized with respect to time and data units. The average length of ON state for a voice source was chosen as the time unit (i.e., $\mu_{vo}^{-1} = 1$), which corresponds to 352 msec. Based on this time unit,

it is assumed that $r_{vo}^{-1} = 1.85$ (or 650 msec), $r_d^{-1} = 4$ and $\mu_d^{-1} = 2$. As noted, a data source is more bursty than a voice source. The size of a cell (i.e., 53 bytes) is used as the data unit. In Figure 4.4, the effect of the size of BNR on the loss probability of NRT cells is shown for two levels of NRT offered load: $\rho_n = 98\%$ (heavy load), and $\rho_n = 64\%$ (light load). Here, $\rho_n \triangleq \overline{\lambda_1}/(1-\gamma)C$. These results were obtained from the analytical model for finite and infinite buffers using $N_d = 3$, $N_{vo} = 6$, $B_2 = 100$, $T = 50$, $\gamma = 0.67$, and $\xi = 0.2$. As noted, the infinite buffer approximation provides a lower bound on the finite buffer performance.

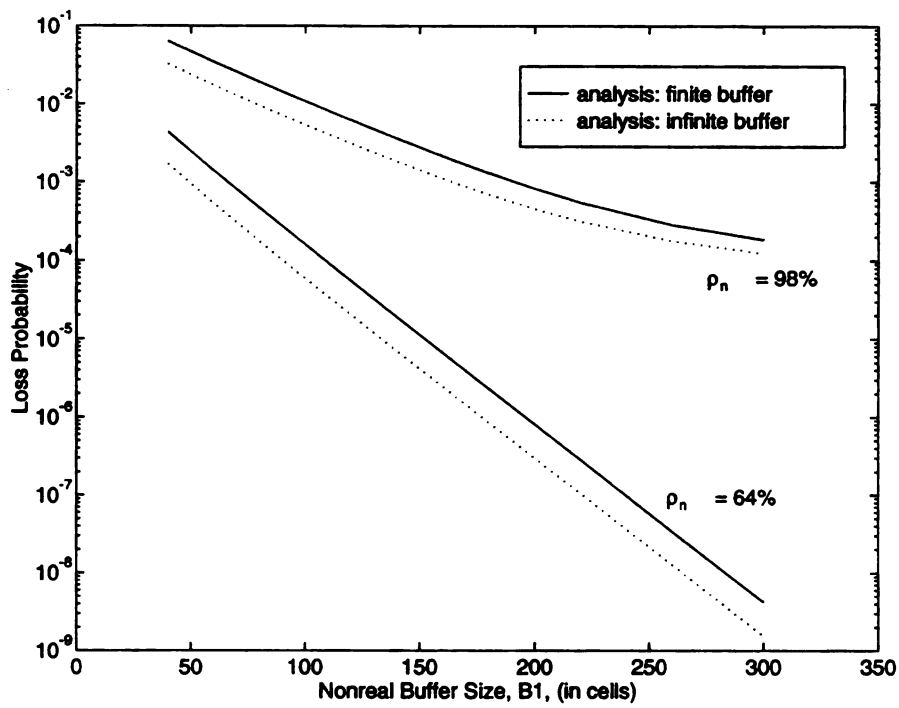

Figure 4.4: Loss probability of NRT cells versus the size of BNR.

The complementary buffer distribution for the content of BR (i.e., $\Pr\{X_2 > x\}$) is shown in Figure 4.5 as a function of x for different values of T. Notice that $\Pr\{X_2 > x\}$, which is based on infinite buffers, approximates the loss probability of RH cells. The confluence of drifts at the threshold level is the reason for the discontinuity of $\Pr\{X_2 > x\}$ at $x = T$.

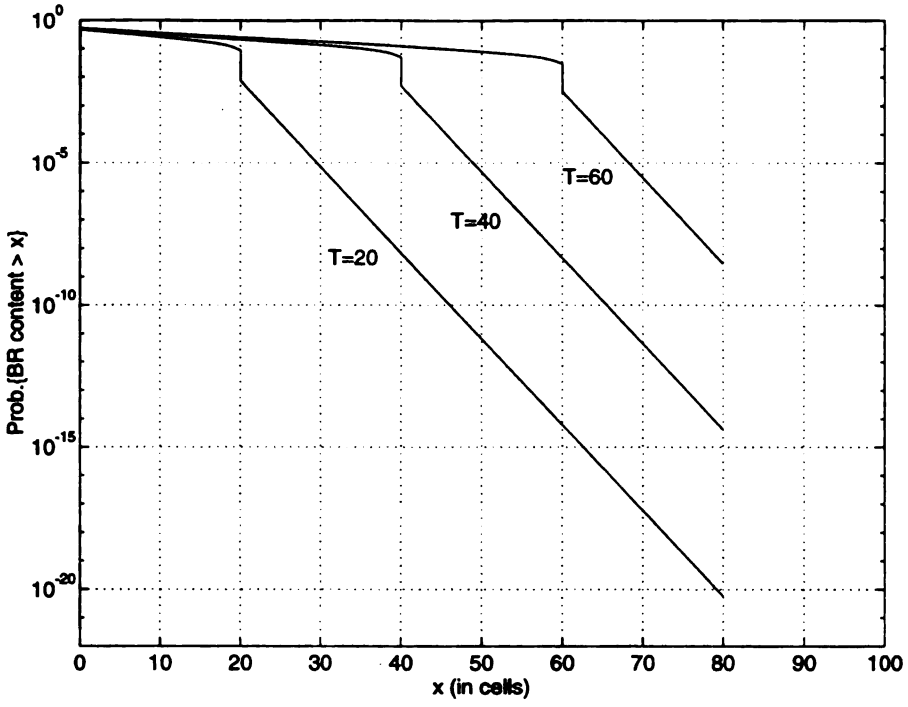

Figure 4.5: Complementary buffer distribution for BR.

The effect of T on the loss probability for RL cells is shown in Figure 4.6 for different values of γ and using $N_d = 3$, $N_{vo} = 8$, $B_1 = 200$, $B_2 = 100$, and $\xi = 0.2$. Results in this figure were obtained using the analytical model with finite buffers. Note that the parameter γ is a key factor is determining the tradeoff between the performance for NRT and RT traffic. This parameter should be determined according to the desired QoS requirements. At larger values of γ, the value of T has more impact on the loss probability of RL cells. For the same experiment, the average waiting times of RH cells are shown in Figure 4.7. At lower values of γ, the value of T has more effect on the delay performance. Compared to the previous figure, at smaller values of γ the waiting time of RH cells is the significant factor in determining T, while at large γ, it is the loss probability of RL cells that determines T.

In Figure 4.8 the fluid approach with infinite buffers was used along with simulations to obtain the loss probability of RL cells and the average waiting time of RH

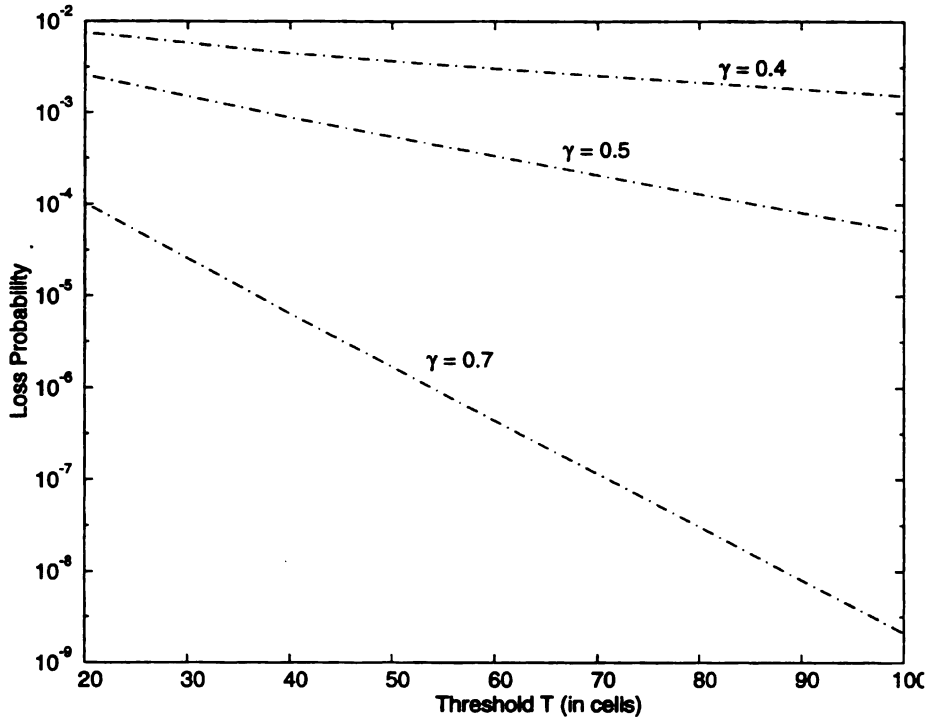

Figure 4.6: Loss probability for RL cells versus threshold.

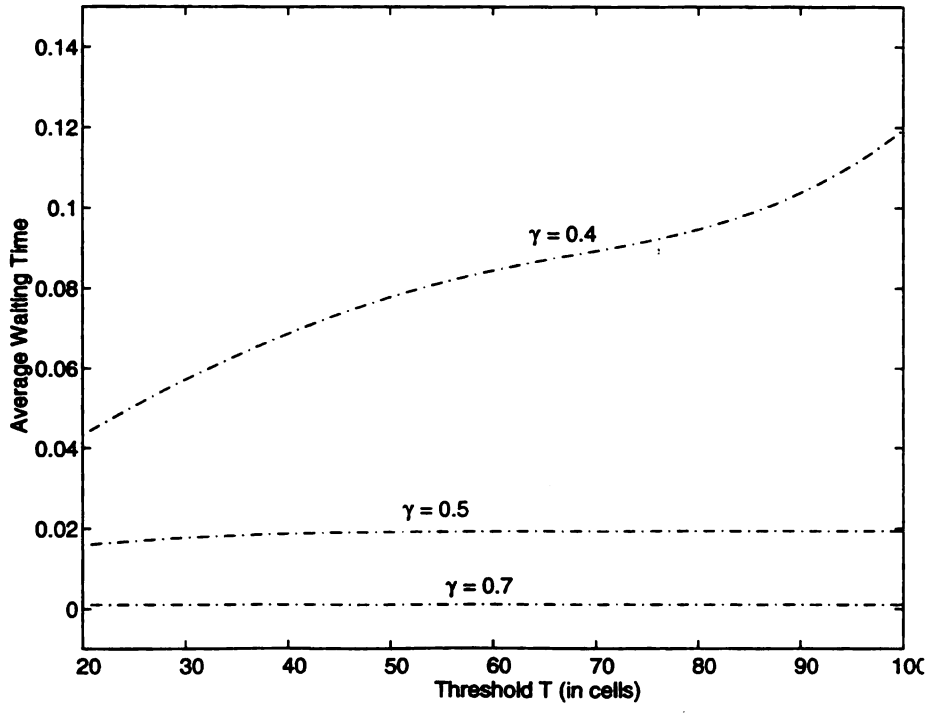

Figure 4.7: Average waiting time for RH cells versus threshold.

cells. The results were obtained for different values of T and using $N_d = 5$, $N_{vo} = 10$, $B_1 = 200$, $B_2 = 80$, and $\xi = 0.2$. Note that the performance predicted by simulations is slightly worse than that obtained from the analysis. For the same experiment, the probability that the waiting time of a real-time cell exceeds a fixed value $t = 0.2841$ time units (100 msec) is shown in Figure 4.9 as a function of T. Again, the analytical results indicate a slightly better performance than the one demonstrated via simulations. The discrepancy between the two approaches, however, is minor. The figure also shows that $\Pr\{W_2^H > t\} > \Pr\{W_2^L > t\}$ which is an intuitive result. The dotted line in the figure indicates the value of T below which the waiting time of an RH or an RL cell will be less than t, with probability one.

The effect of the load ratio of RT traffic (ξ) on the performance is illustrated in Figures 4.10 and 4.11. Three values of T were used: 20, 40, and 60. Figure 4.10 shows that the loss probability of RL cells increases with the load ratio. This is indicated by the analytical (with infinite buffer) and simulation results. The same trend can be seen in Figure 4.11 where the average waiting time of a RH cell is shown to increase with ξ (only analytical results are shown here). The implication of Figures 4.10 and 4.11 is that a threshold-based cell discarding mechanism such as NTCD is most effective when ξ is small, i.e., when the high priority cells constitute a relatively small fraction of the traffic.

4.8 A Note on Fluid Models for Video Traffic

In previous sections we used the fluid approach to analyze the performance of NTCD/MB under voice and data streams. The analysis can be extended to include any type of traffic, as long as this traffic can be adequately characterized by a MMFF process. In this section, we describe how the previous analysis of NTCD/MB can be extended to include fluid models for video traffic. These models, which were proposed

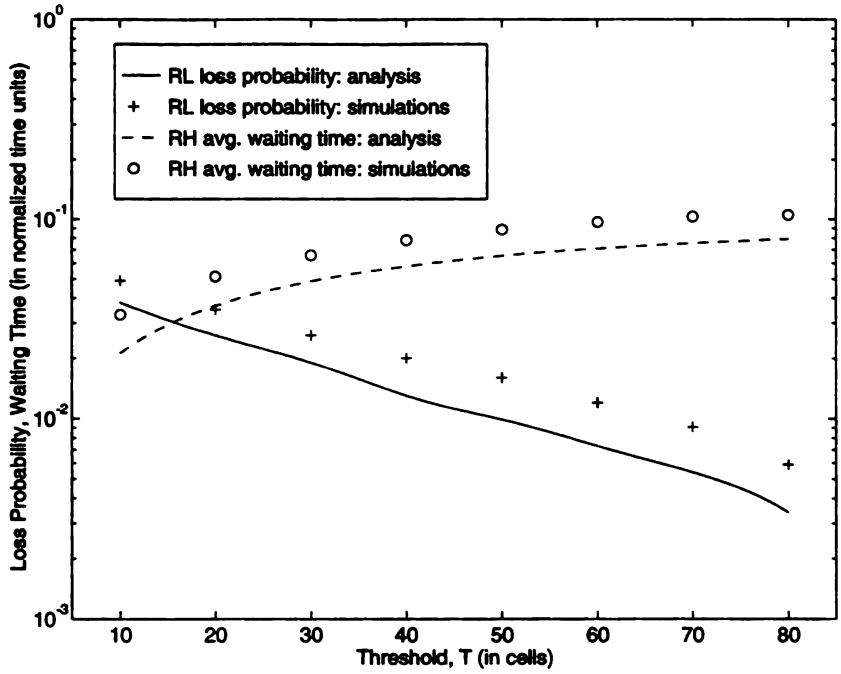

Figure 4.8: Loss probability for RL and average waiting time for RH versus threshold.

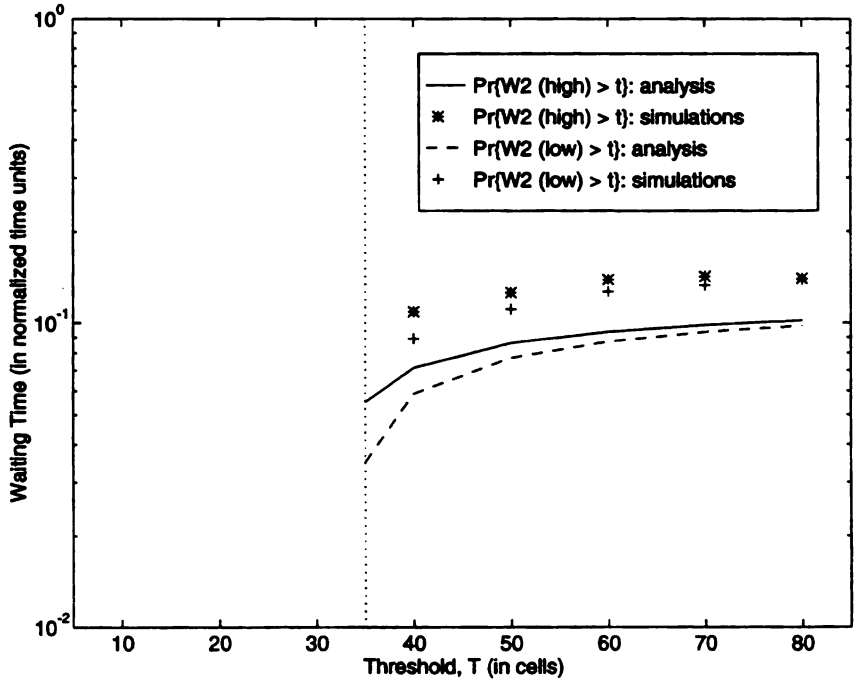

Figure 4.9: Probability that the waiting time for RT cells exceeds $t = 0.2841$.

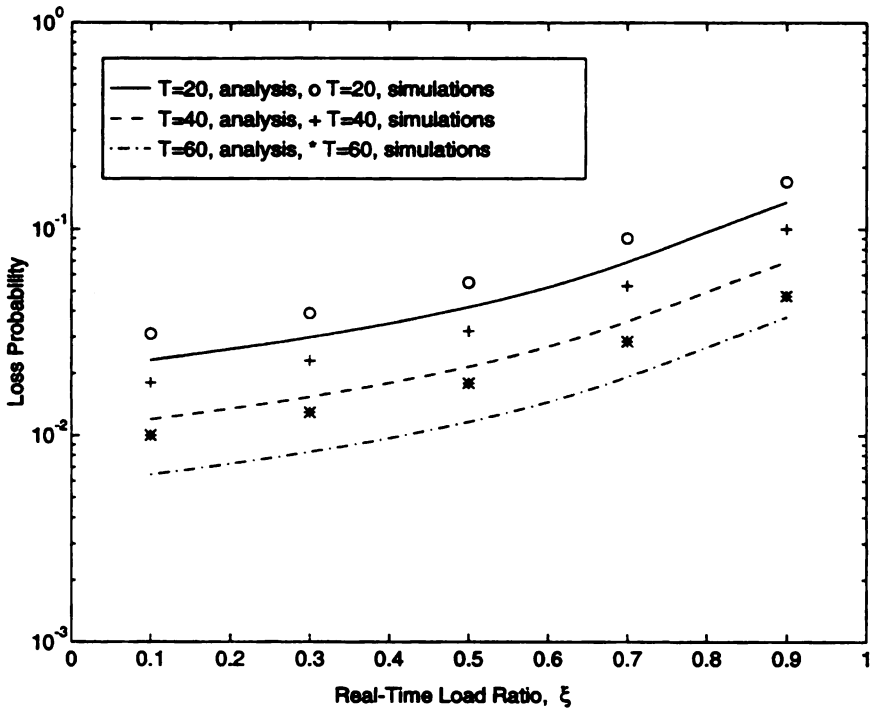

Figure 4.10: Loss probability for RL cells versus RT load ratio.

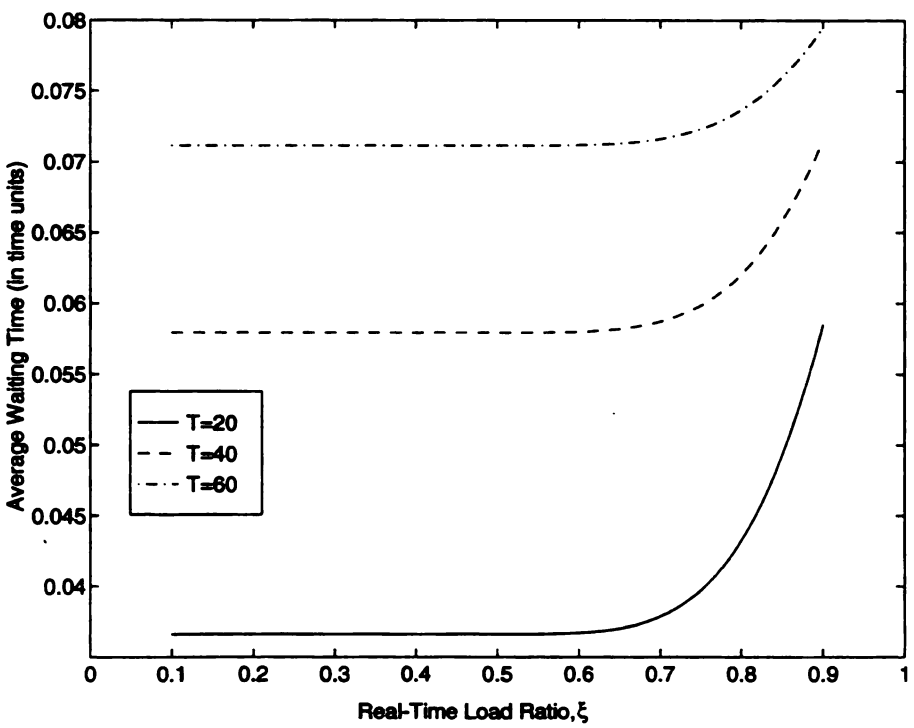

Figure 4.11: Average waiting time for a RH cell versus RT load ratio.

in [52] and [62], are applicable *only* to video streams generated by video-phones, and which are compressed based on the conditional replenishment compression algorithm (similar to differential PCM coding).

Maglaris *et al.* [52] analyzed video streams generated by video-phones and developed two traffic models for these streams. One of these models, which we adopt in this section, is a fluid model. In this model, the aggregate bit rate of N_{vd} video sources is quantized into a number of discrete levels, as shown in Figure 4.12.

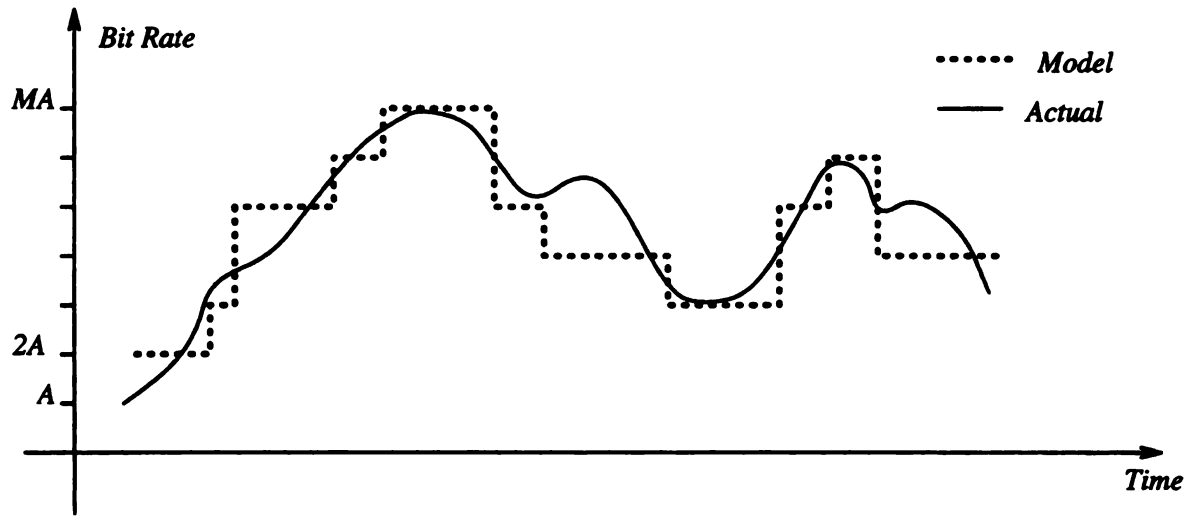

Figure 4.12: Aggregate arrival rate from N_{vd} video sources.

The quantized aggregate bit rate constitutes a Markov chain. Transitions between states follow the birth-death transition diagram shown in Figure 4.13. At state i, the total arrival rate is $\lambda_{vd}(i) = Ai$, $i \in \mathcal{L}_{vd} \triangleq \{0, 1, \ldots, M\}$, where A is the quantization step (difference between two successive levels) and M is the number of quantization levels. Transitions occur between adjacent states only (i.e., arrival rate increases gradually). A transition from state i to state $i+1$ occurs at an average rate of $(M-1)r_{vd}$. Likewise, the average transition rate from state i to state $i-1$ is $i\mu_{vd}$. The process has more tendency to go to a lower level at high rates, and to a higher level at low rates. Let the generator matrix and the equilibrium probability vector be denoted by Q_{vd} and Π_{vd}, respectively. The expressions for the two quantities are

similar to the ones for Q_1 and Π_1. The values for r_{vd} and μ_{vd} are found by matching the mean, standard deviation, and the autocorrelation function in both the model and the empirical data. In [52] the values of r_{vd}, μ_{vd} and A were given in terms of N_{vd} and M. It was noted that for large M the results are insensitive to the particular choice of M. A value of $M = 20N_{vd}$ was recommended, as it agreed with results from another simulation model.

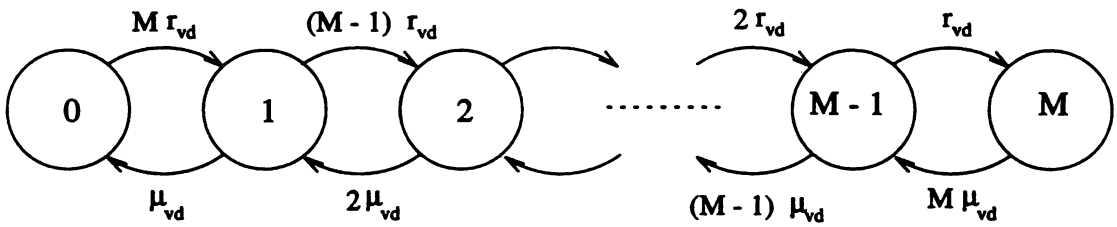

Figure 4.13: Transition diagram for the fluid model of video traffic.

Consider the structure of NTCD/MB that was shown in Figure 4.2. In addition to voice and data streams, video streams are also incorporated within RT traffic. Similar to voice streams, video streams consist of RH and RL cells. The aggregate video traffic is modeled as an $(M+1)$-state MMFF. Independence of voice and video sources allows us to describe both traffic types as one $(N_{vo}+1)(M+1)$-state MMFF process whose state $S^*(t)$, at time t, represents a 2-tuple integer which consists of the state of voice traffic, and the state of video traffic.

$$\begin{aligned} S^*(t) \in \mathcal{L}^* &\triangleq \{0, 1, \ldots, K^*\} \\ &= \{(i,j) : i \in \mathcal{L}_{vo},\ j \in \mathcal{L}_{vd}\} \\ &= \{(0,0),(0,1),\ldots,(0,M),(1,0),\ldots,(1,M),\ldots,(N_{vo},M)\} \end{aligned} \tag{4.35}$$

where $K^* = (N_{vo}+1)(M+1) - 1$. It is important here to note the lexicographic ordering of the indices in (4.35). A general treatment of large systems which are composed of several separable subsystems is found in [66]. Two processes are separable if they are independent and are, individually, modulated by a reversible Markov

process. Using results from [66], the combined generator matrix for real traffic is the $K^* \times K^*$ matrix

$$\boldsymbol{Q}^* = \boldsymbol{Q}_{vo} \oplus \boldsymbol{Q}_{vd} \tag{4.36}$$

The equilibrium probability vector for the process that characterizes RT traffic can be expressed as

$$\Pi^* = \Pi_{vo} \otimes \Pi_{vd} \tag{4.37}$$

In the above, "$\oplus$" and "$\otimes$" indicate the Kronecker sum and Kronecker product, respectively. The total arrival rate for real-time traffic at state i is given by

$$\lambda_2(i) = \lambda_{vo}\left(\left\lfloor \frac{i}{M} \right\rfloor\right) + \lambda_{vd}\,(i \text{ modulo } M) \tag{4.38}$$

$$= \alpha_{vo}\left\lfloor \frac{i}{M} \right\rfloor + A\,(i \text{ modulo } M) \tag{4.39}$$

The function $\left\lfloor \frac{x}{y} \right\rfloor$ indicates the integer part of the division. Without loss of generality, we assume that $\alpha_2 \neq nM$ for any integer n, so that the arrival rates at two different states in $\mathcal{L}^*$ are always distinct. The load ratio ξ which was defined for voice traffic applies as well for video traffic, so that $\xi = \lambda_2^H(i)/\lambda_2(i)$.

The combined total arrival rate of NRT traffic and RT traffic (which now consists of voice and video streams) can be described by a global K-state MMFF process ($K = (N_d + 1)(N_{vo} + 1)(M + 1)$) whose state at time t is $S(t) \triangleq (S_d(t), S^*(t)) \in \mathcal{L}$, where $\mathcal{L} \triangleq \{(i,j) : i = 0, 1, \ldots, N_d,\ j = 0, 1, \ldots, K^*\}$. The set of pairs in $\mathcal{L}$ are ordered lexicographically. We say $S(t) = (i,j) = k$ where $k = 1, 2, \ldots, K$, to mean that the integer k represents the k'th ordered pair in $\mathcal{L}$. The global generator matrix $\boldsymbol{Q}$ and the stationary probability vector Π for the global arrival process are [66]:

$$\boldsymbol{Q} = \boldsymbol{Q}_d \oplus \boldsymbol{Q}^* \tag{4.40}$$

$$\Pi = \Pi_d \otimes \Pi^* = [\pi_1\ \pi_2\ \ldots\ \pi_K]^T \tag{4.41}$$

Once the aggregate traffic is described as a MMFF process, the analysis of NTCD/MB discussed in previous sections can be applied in a straightforward manner.

The underlying assumption in the analysis developed in this chapter is that a traffic stream can be adequately characterized by a fluid model. While such an assumption is quite acceptable in the cases of data and voice streams, it is unlikely that this assumption is appropriate for *every* compressed video stream. The fluid models proposed in [52] and [62] are applicable *only* to certain types of video (i.e., video-phones) and based on certain compression schemes (i.e., the conditional replenishment scheme). In fact, it is quite improbable that a single traffic model can adequately characterize all sorts of compressed video streams due to two reasons. First, different types of video are quite different with regard to their scene dynamics (e.g., video-phones depict much less scene activity than video movies, which in turn depict less activity than commercial advertisements). Second, the existence of various compression algorithms result in traffic streams with different patterns, i.e., the same (uncompressed) video source can be compressed in different ways, resulting in different traffic patterns.

Fortunately, a new standard for video compression has emerged and is expected to receive a wide distribution. This standard is known as *Moving-Pictures Expert Group* (MPEG) [26, 27] (after the name of the ISO committee which was responsible for its development). In order to investigate the performance of NTCD/MB under MPEG-coded video traffic, models for MPEG streams must first be developed. This is the focus of the next chapter.

Chapter 5

Characterization of MPEG-Coded Video Streams

5.1 Introduction

The largest proportion of the bandwidth in B-ISDN/ATM networks will be used to transport video streams. This is due to the introduction of many new video/multimedia services and also to the large amount of bandwidth needed to transport real-time digital video. The huge link capacity provided by optical fibers can be quickly saturated by few video streams transmitted in their digital format. Hence, compression of digital video is the only means to transport real-time full-motion video over B-ISDN/ATM networks. By means of compression, the number of simultaneously transported video streams over a single link can be increased to more than one order of magnitude. The basic concept in compression is to reduce the size of video frames by taking advantage of the inherent information redundancy (spatial as well as temporal) that exists in video data. A video stream (before compression) consists of a sequence of frames (i.e., images) that are displayed successively at a constant rate (e.g., 30 frames/sec in the NTSC standard) to produce motion picture . Most

often successive frames exhibit a large amount of visual similarities. This fact is used by many compression schemes to encode the differences between frames only. Other types of compression can also be used such as discrete cosine transform (DCT), motion prediction, and interpolation. The size of a compressed video frame varies from one frame to another depending on the scene activity and the type of compression. As a result, the output of a video compressor is a variable bit rate (VBR) stream. Unlike conventional data networks, ATM networks provide efficient support for VBR traffic.

Since this research is concerned with the design and analysis of the NTCD/MB mechanism under diverse types of traffic, we would like to study the performance of NTCD/MB under compressed video traffic. Such a study, however, requires the availability of empirical video data to be used in trace-driven simulations, or traffic models to characterize the actual video streams. Obtaining empirical data is a difficult problem due to the amount of storage which would be required to store a digitized movie before compression (a 2-hour movie with a moderate level of activity would require in the order of few hundreds of gigabytes). Developing a traffic model for compressed video is also a challenging problem, due to both the difficulty in obtaining empirical data, and the complexity of the traffic patterns of compressed video streams. While a number of traffic models have been proposed to characterize compressed video, these models are valid *only* for a particular type of video (e.g., video-phone, video-teleconferencing) based on a particular compression technique. Hence, none of the existing models are considered to be universal for any compressed video stream.

We focus on the modeling and characterization of video streams that are compressed based on the Moving Picture Expert Group (MPEG) compression technique. MPEG is a new compression standard which has recently gained a considerable attention. The first draft for the MPEG standard was accomplished in 1991 [26]. This draft defines the compressed bit stream for video and audio within a bandwidth of

1.5 Mbits/s. A more recent draft (known as MPEG-2) [27] defines the compression sequence at the rate of 5 Mbits/s. MPEG-2 has just been accepted as the basis for the HDTV standard in the United States [1]. The standardization and widespread acceptance of MPEG are the main reasons for choosing it among other compression techniques to be the focus of our research. In addition to these merits, MPEG involves several modes of compression, some of which are the basis of other (non-standard) compression algorithms. Thus, a traffic model for an MPEG stream would be applicable, as well, to video streams that are compressed using other compression algorithms (for example, JPEG which is a still-image compression algorithm is already being incorporated in MPEG).

Several traffic models have been proposed to characterize compressed video streams [52, 62, 21, 54, 24, 61, 63, 46, 74, 53, 6, 7, 15]. The parameters of these models were obtained by matching certain statistical characteristics of an actual video sequence to their counterparts in the studied model. Particular emphasis was on matching the correlation structure of the bits-per-frame sequence, since correlations are known to have a great impact on the queueing performance of a statistical multiplexer [65]. Correlations arise naturally in video traffic and their patterns are largely determined by the type of video (e.g., video-phone, motion-picture, teleconferencing) as well as by the compression algorithm used. In [52] the authors proposed two traffic models for the frame size sequence of a video-phone stream compressed using a conditional replenishment algorithm. The first model is a first-order autoregressive process, while the second is a fluid model. A more elaborate model based on an ARMA process was proposed in [21]. Skelly *et al.* [63] introduced a histogram-based traffic model using a quasi-static approximation. In [46] the authors used the TES approach developed in [54] to model a video stream that was generated using DCT coding, but with no differential or motion compensation. Heyman *et al.* [24] analyzed the VBR trace for a 30-minute long video-teleconferencing sequence and suggested

that the number of cells per frame approximately follows a gamma distribution. A sophisticated autoregressive model was proposed in [61], which consists of the sum of two AR(1) processes and a Markov chain. More recently, some researchers investigated the appropriateness of fractional ARIMA processes to capture long-range dependency in the autocorrelation function (acf) of the VBR stream [17]. We see this last approach quite promising to model MPEG-compressed streams. However, the video sequence used in [17] was obtained using a DCT-based scheme that does not exploit the various compression modes employed in the MPEG algorithm. An empirical study of the characteristics of several short sequences of MPEG-compressed video was reported in [58]. No traffic models were provided in that study.

Our objectives in this chapter are twofold. First, we describe the characteristics of an MPEG-coded video movie based on empirical data. Second, we propose two traffic models for MPEG streams. A video movie exhibits more scene dynamics and is, therefore, more difficult to analyze than other types of video such as video-conferencing and video-phones. With regard to the first goal, we describe an experimental setup which was used to capture, digitize, and compress a long portion of a video movie. The frame size sequence of the compressed movie was analyzed. Empirical statistics of this sequence were obtained and subsequently used in developing traffic models. Such a sequence (which we refer to as the VBR sequence) was also used in trace-driven simulations described in *Chapter 5*.

5.2 Overview of the MPEG Standard

This section provides a brief overview of the MPEG standard. The description is related to the first draft of MPEG, known as MPEG-I [26]. Subsequent drafts (e.g., MPEG-II) have slight differences which will not be discussed here. Most of the material in this section is paraphrased from [47]. Further details can be found in [26]

and [27].

In response to the need for standardization of video compression techniques, the International Organization for Standards (ISO) undertook an effort to develop a standard for video and associated audio on digital storage medium which include CD-ROM, tape drives, and writable optical drives, as well as telecommunication channels such as ISDNs, and local area networks. This effort resulted in the MPEG standard, named after the ISO group that was responsible for its development. The MPEG activities cover more than video compression, since the compression of the associated audio and the issue of audio-visual synchronization cannot be worked independently of the video compression. MPEG-Video is addressing the compression of video signals at about 1.5 Mbits. MPEG-Audio is addressing the compression of a digital audio signal at the rates of 64, 128, and 192 Kbits/s. MPEG-System is addressing the issue of synchronization and multiplexing of multiple compressed streams. Our focus is on MPEG-Video compression. During the standardization phase, the MPEG committee took into consideration the related activities of other standard organizations. As a result, the MPEG standard incorporated many of the good features of other compression standards.

5.2.1 Features of the MPEG Compression Algorithm

Since the ISO/MPEG committee consists of various segments of the information processing industry, a representation of video on digital storage media has to support many applications. This is expressed by saying that the MPEG standard is a *generic* standard. It is generic in the sense that it is independent of a particular application. This does not mean that the standard ignores the requirements of the applications. A generic standard possesses certain features that make it somewhat universal. Some of the important features of the MPEG compression standard that were identified to meet the needs of applications are:

- *Random Access.*

  This feature is essential for video on a storage medium whether or not the medium is a random access medium such as a CD, or a sequential medium such as a magnetic tape. Random access implies the existence of access points (i.e., segments of information coded only with reference to themselves).

- *Fast Forward/Reverse Searches.*

  Depending on the application, it should be possible to scan a compressed bit stream and, using the appropriate access points, display selected pictures to obtain a fast forward or reverse effect.

- *Reverse Playback.*

  Some interactive applications require the video signal to play in reverse. This feature should be possible with slight additional cost in memory (at the expense of slightly less quality of video when the signal is played back).

- *Coding/Decoding Delay.*

  Since video quality and coding/decoding delay represent a trade-off to a certain extent, the MPEG algorithm is required to perform well over the range of acceptable delays.

The MPEG compression algorithm relies on two basic techniques: block-based motion compensation for the reduction of the temporal redundancy and DCT-based (discrete cosine transform) compression for the reduction of the spatial redundancy. Motion compensation techniques are applied with both causal (i.e., predictive) and noncausal (i.e., interpolative) coding.

5.2.2 Temporal Redundancy Reduction

To achieve significant bit-rate reduction, temporal redundancy is exploited by generating three types of compressed frames (i.e., images): Intra-coded (I), Predictive

(P), and Bidirectional (B) frames. I frames provide access points for random access, with moderate compression. An I frame is encoded independently of other frames based on DCT coding. A P frame is encoded with reference to a past I or P frame, and is used as a reference point for the next P frame. A B frame is an interpolated frame that requires both a past and a future reference frames (I or P), and it results in the highest amount of compression. B frames are never used as reference frames. From the point of view of the encoder, a stream of frames could look like the pattern shown in Figure 5.1-a. Since B frames require noncausal encoding, the transmission sequence has a different pattern than the encoding sequence, as shown in part (b) of the figure.

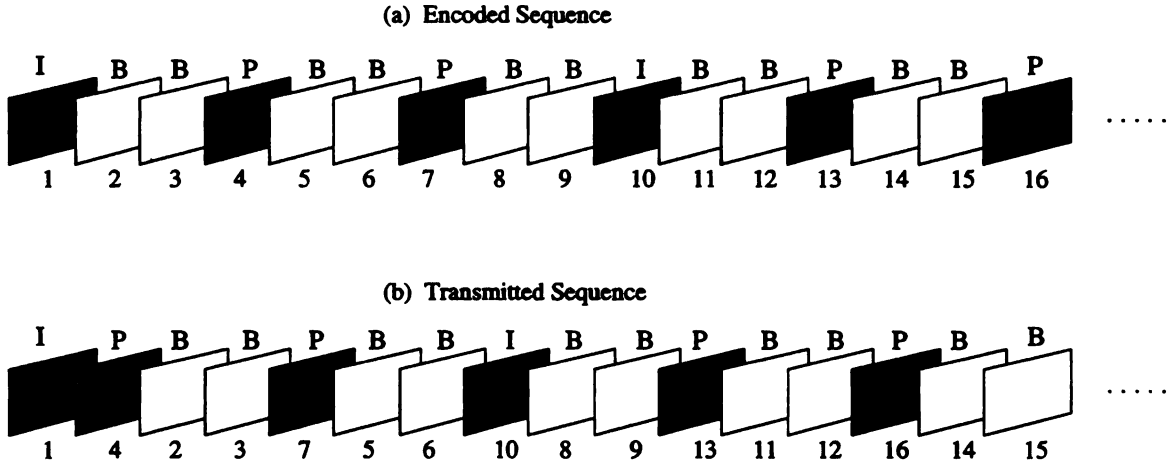

Figure 5.1: A sequence of MPEG frames.

At the encoder side (part (a) of the figure), the sequence of frames are generated according to a predetermined pattern which we refer to as the *compression pattern.* This pattern is repeated periodically until the whole stream is compressed. The pattern defines the numbers and types of P and B frames between successive I frames. In Figure 5.1, the pattern is IBBPBBPBB which continues to be repeated until the end of the video stream. We will refer to the length of this pattern as the *compression period.* The transmitted sequence (part (b) of Figure 5.1) is a permutation of the ordering of the original sequence. If we only consider the types of the frames (and

not their time order), except for the first few frames, the encoded and transmitted sequences look alike. Such an observation will be used later when discussing the autocorrelation function of the transmitted sequence.

5.2.3 Spatial Redundancy Reduction

Both still-image and prediction error signals have a very high spatial redundancy. Still-image refers to I frames and portions of P and B frames that are intra-coded. Prediction error refers to the difference between the predicted frames (produced using motion estimation techniques) and the actual frames. Because of the block-based nature of motion compensation process, block-based spatial redundancy reduction techniques are used. These techniques include transform coding and vector quantization coding. Similar to JPEG (Joint Photographic Experts Group) and CCITT H.261 standards, the MPEG standard was based on DCT transform. Intra-frame coding with DCT involves three stages: computation of the transform coefficients; quantization of these coefficients; and performing run-length encoding on the quantized coefficients.

5.3 The Experimental Setup

In order to provide a meaningful statistical study of an MPEG-coded stream, a long sequence of actual video is needed. We used the the experimental setup described below to capture, digitize, and compress a 23-minutes long video sequence [42]. The sequence was taken from the entertainment movie *The Wizard of Oz*, and was digitized and encoded using a public domain software MPEG encoder developed by the Berkeley Plateau Research Group.

A two-stage approach was used to obtain the VBR trace (i.e., the sequence of bits per frame in the compressed stream). In the first stage, video frames were digitized

and stored on magnetic tape. A computer program was written to capture and digitize video frames. The program ran on a workstation connected to the laser disk player (see Figure 5.2). Using a *SunVideo* card installed in the workstation, the program received analog frames and digitized them one at a time. It then sent control signals to the laser player to position the head at the next frame and repeat the process. Frames were converted to 4:1:1 YUV format (the input format for MPEG encoders) and were written to hard disk. After accumulating some number of frames (180 frames), these frames were moved from the hard disk to magnetic tape. The process continued until the entire stream was stored on tape. A pseudo-code for the program that was used to obtain the digitized frames is shown in Figure 5.3.

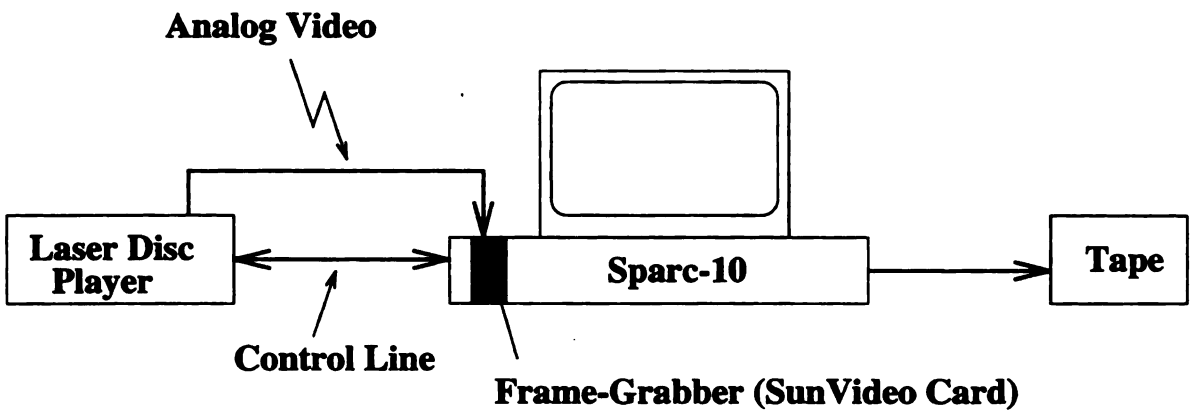

Figure 5.2: A schematic diagram of the experimental setup.

In the second stage, the frames were read from tape and compressed using the Berkeley MPEG encoder. The VBR trace was extracted during the encoding process. The compression pattern used to encode the examined video stream is shown in Figure 5.4. In this pattern, the compression period is 15, while the distance between successive *P* frames within the compression period is 3. There is no particular reason for choosing such a pattern except that it is one of the commonly used patterns (empirically, this pattern is found to provide good compromise between complexity and the amount of compression). In fact, the MPEG standard does not specify the compression pattern to be used, which makes the compression pattern a user-

```
Procedure: Main
begin
   position tape
   for i = 1 to Last_Set
      call Digitize(i)
      store digitized frames on tape
   end for
end


Procedure: Digitize
begin
   open a serial port
   configure video card (link to XIL Lib.)
   position player head (random mode)
   start player
   while < 180 frames are captured
      grab a frame (using XIL functions)
      convert frame to YUV format
      write frame to hard disk
      position player head (sequential mode)
   end while
   stop player
end
```

Figure 5.3: Pseudo-code for the program used in digitizing video frames.

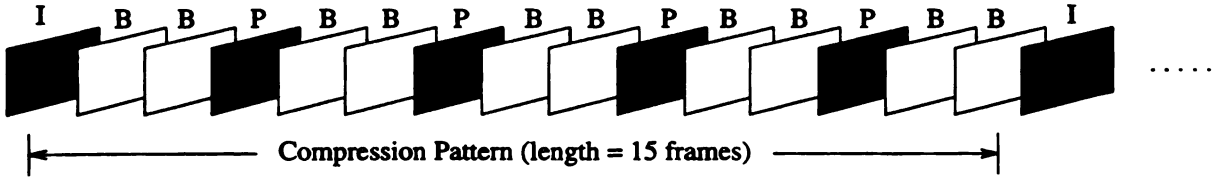

Figure 5.4: Compression pattern used to generate the empirical data.

dependent parameter.

5.4 Empirical Statistics

5.4.1 Frame-Size Measurements

Based on the previous setup, 41760 video frames were compressed and their sizes were recorded. At 30 frames/sec, this represents about 23 minutes of real-time full-motion video. Portions of the frame size sequence are plotted in Figures 5.5 and 5.6. In both figures, the larger frame sizes (which are depicted by the lightly shaded areas) correspond to I frames. The darker parts of the figures correspond to P and B frames which are smaller in size than I frames. Short-range variations in the VBR sequence are shown in Figure 5.7. Sample statistics for the size of an arbitrary frame and also for the sizes of individual I, P, and B frames are given in Table 5.1.

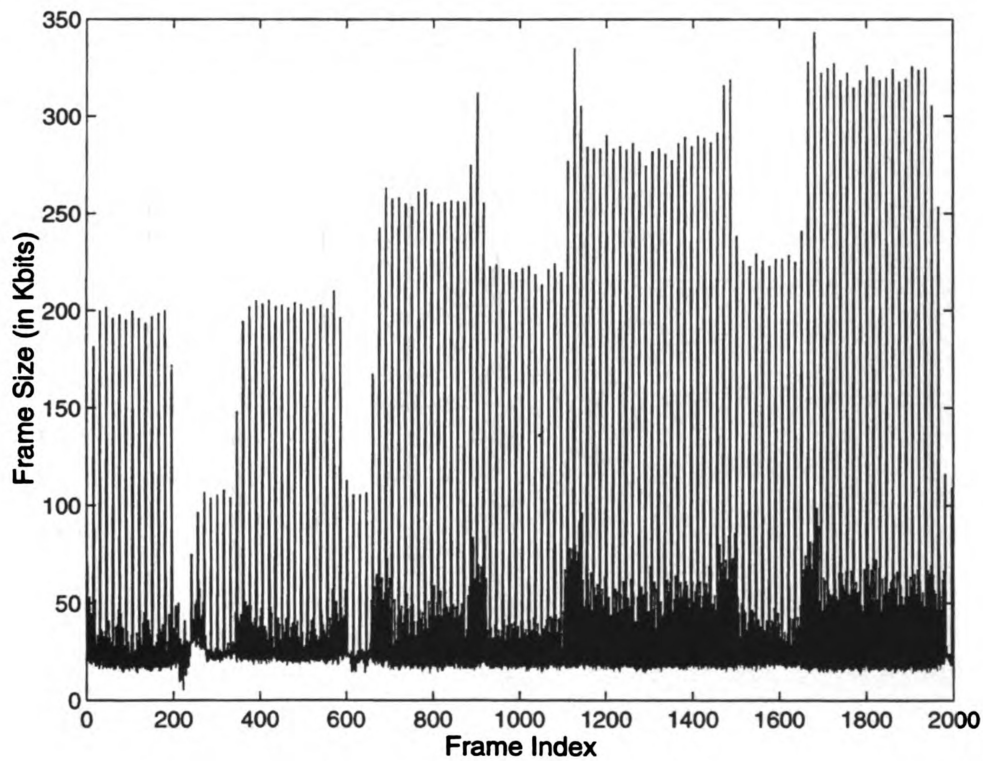

Figure 5.5: Frame size sequence (frames 1 to 2000).

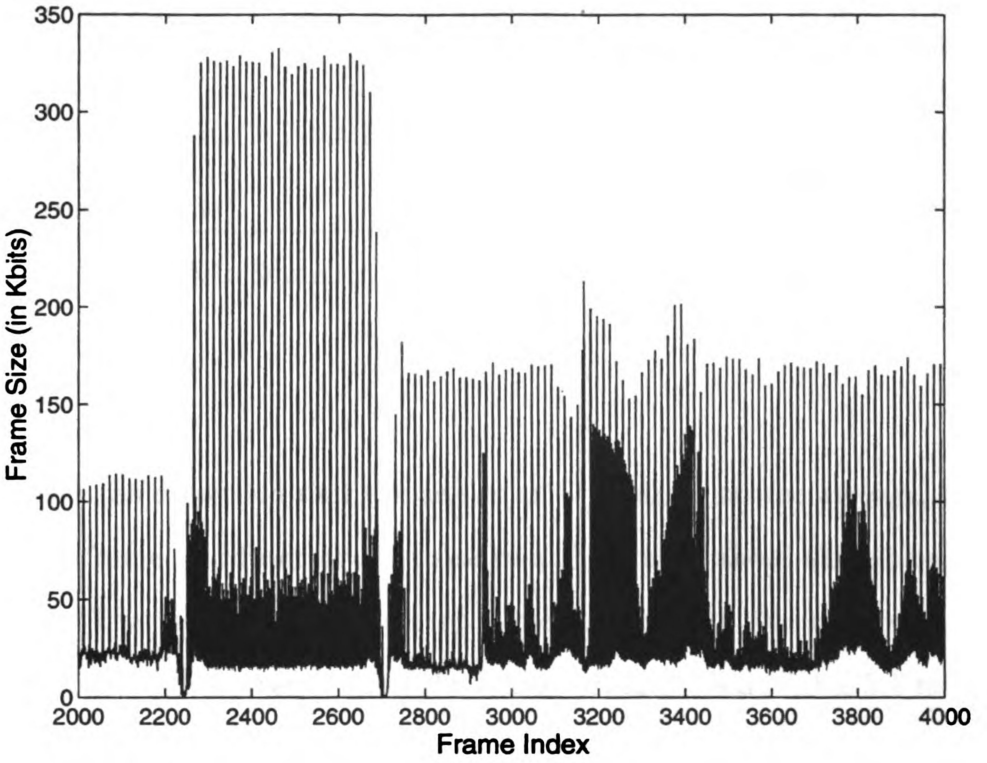

Figure 5.6: Frame size sequence (frames 2001 to 4000).

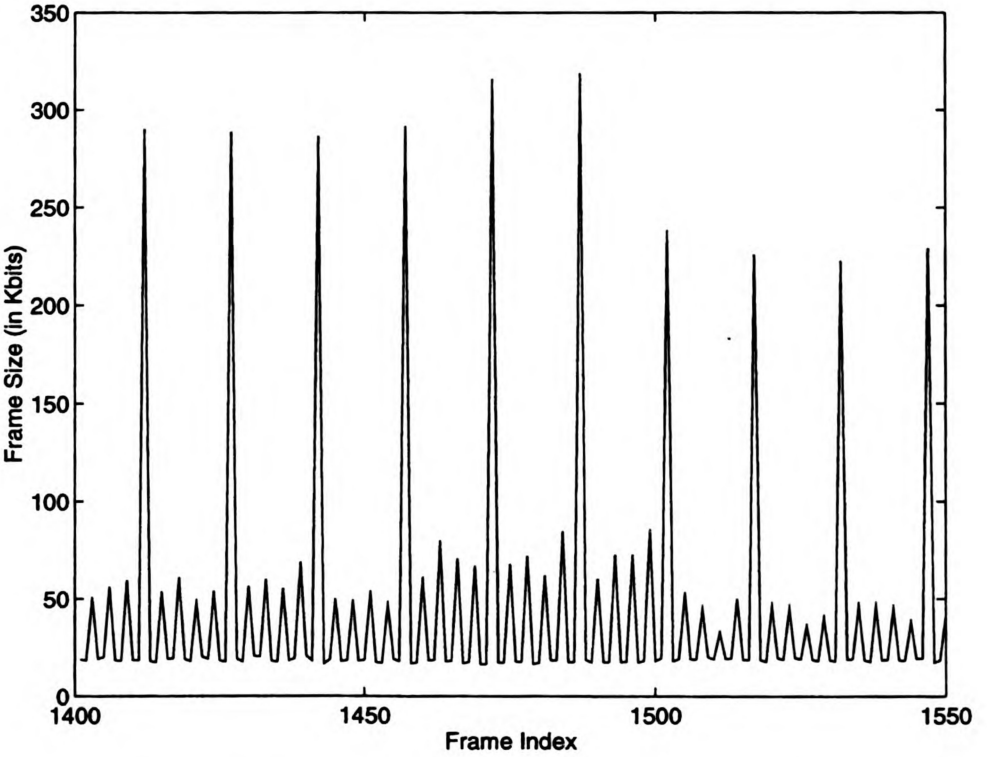

Figure 5.7: Frame size sequence (frames 1401 to 1550).

Sample Statistic	All Frames	I Frames	P Frames	B Frames
Number	41760	2784	11136	27840
Mean (in Kbits)	41.7	197.1	58.0	19.6
Standard Deviation (in Kbits)	51.7	63.8	37.3	5.7
Maximum (in Kbits)	343.1	343.1	284.6	60.0
Minimum (in Kbits)	0.56	36.2	0.56	0.57

Table 5.1: Summary statistic for the frame size sequence.

From the table, we find that the size of an I frame is, on the average, ten times that of a B frame. Additionally, the size of a P frame is about three times that of a B frame. The ratio of the sample standard deviation to the sample mean is an indication of the relative variation of the size of a frame. Using Table 5.1 to compute this ratio, it appears that P frames have the largest relative variation in their sizes followed by I frames, and finally B frames.

The frame size histogram is depicted in Figure 5.8. It has the general shape of a Pareto or a Pearson type V *pdf*. We will not, however, attempt to fit a theoretical *pdf* to this histogram, because such a *pdf* would not differentiate among the three types of frames. Notice that the values in Figure 5.8 are given in cells/frame, since video bits must be converted into cells when entering the network. If the size of a frame at the output of the encoder is X bits, then this frame is packetized into $\lceil X/(8 * 48) \rceil$ cells (i.e., a frame contains an integer number of cells).

It is apparent that the type of a frame is an important factor in determining the size of that frame. Hence, we will be interested in studying the frame size statistics of individual types of frames. For this purpose, we divide the complete VBR trace into three subsequences, each of which corresponds to the size of same-type frames. Sample paths from these subsequences are depicted in Figures 5.9-5.11.

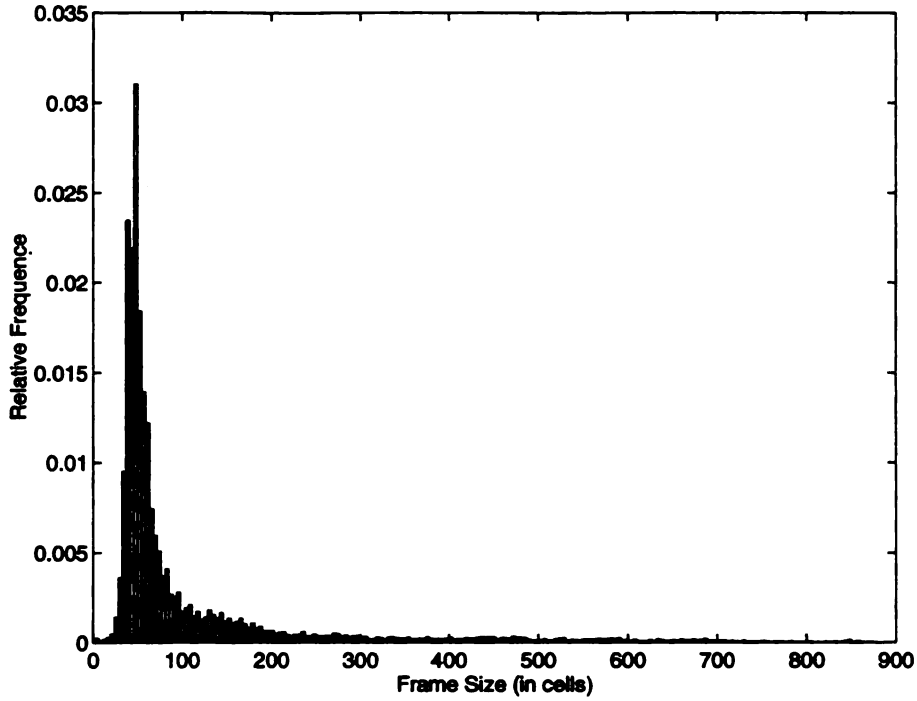

Figure 5.8: Frame size histogram.

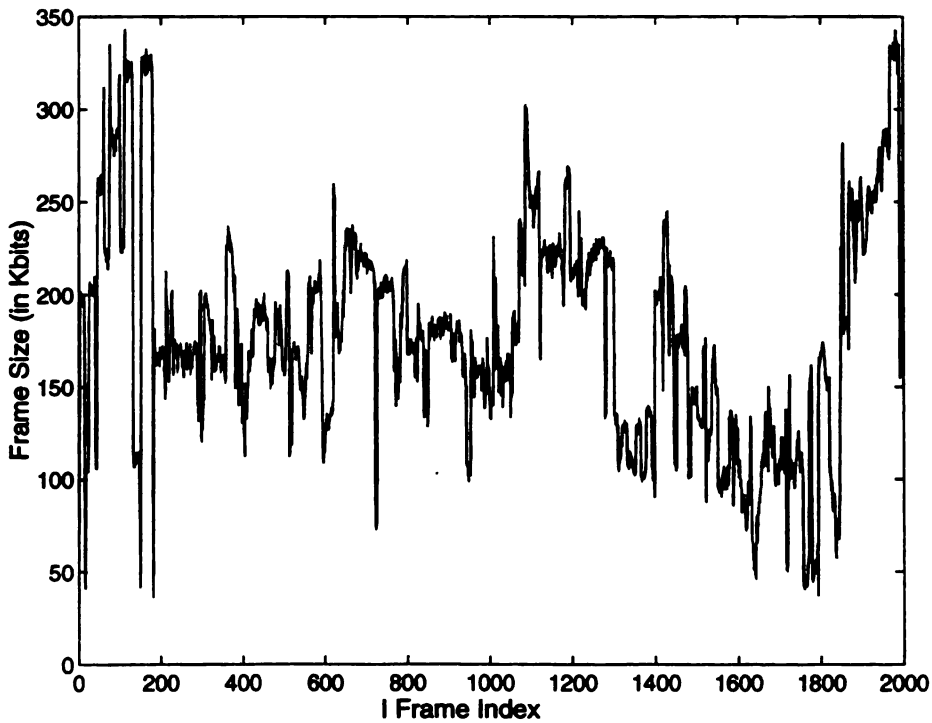

Figure 5.9: Sample path from I-frames subsequence.

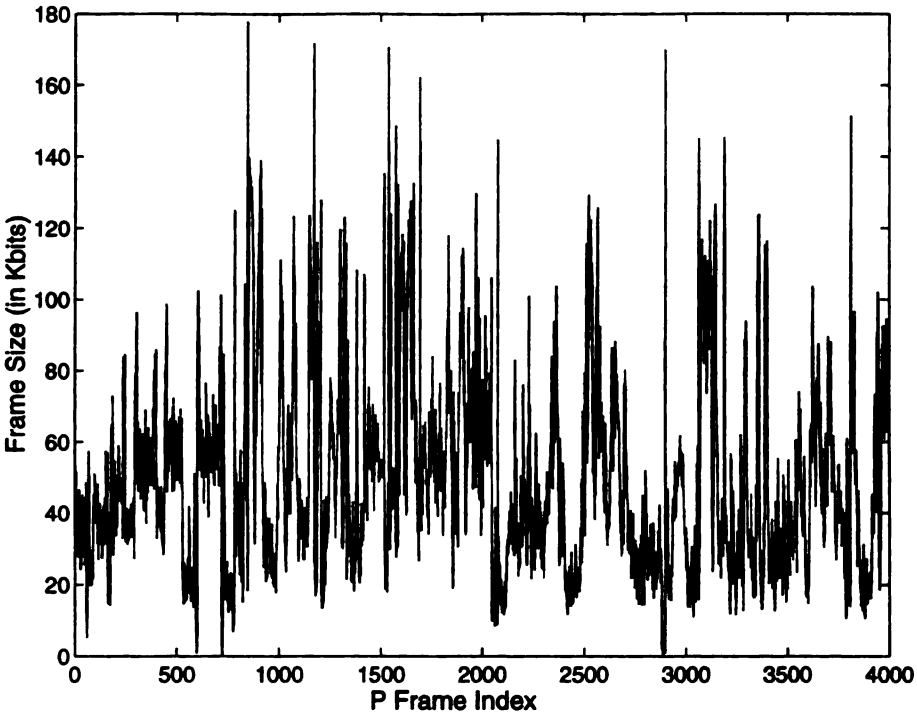

Figure 5.10: Sample path from P-frames subsequence.

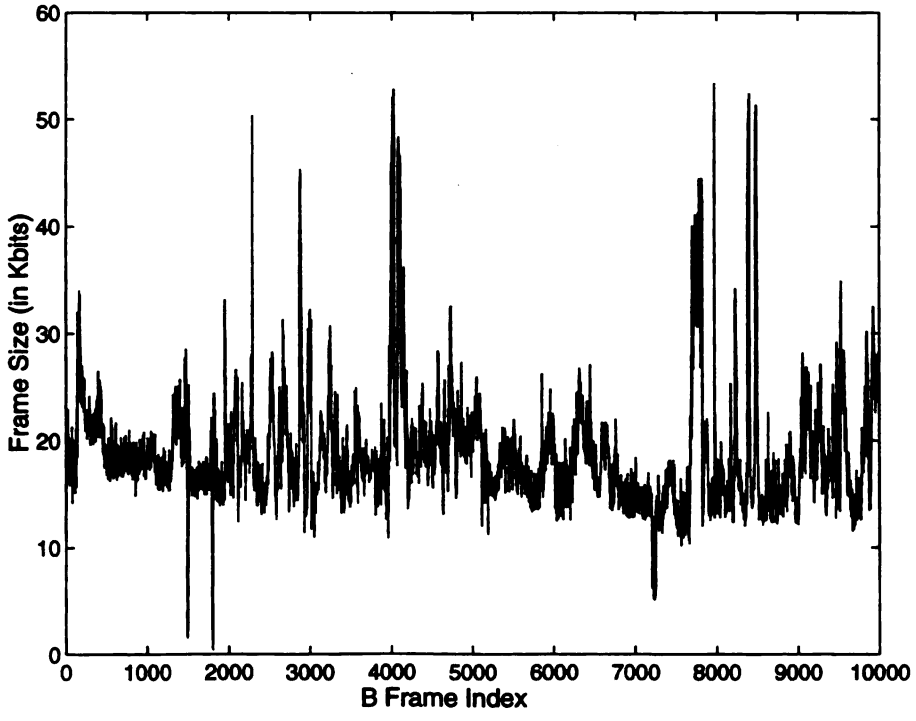

Figure 5.11: Sample path from B-frames subsequence.

5.4.2 The Correlation Structure

Correlations in video data arise as a consequence of visual similarities between consecutive images (or parts of images) in a video stream. Although compression results in reducing the amount of correlations in video, the VBR sequence of a compressed stream maintains a considerable amount of correlations. This is due to the periodic fashion in which a given compression technique is used. For example, consider two consecutive frames that correspond to two semi-similar images. Applying DCT to these frames (one at a time) will result in two compressed frames with almost similar sizes (i.e., highly correlated frame sizes).

Two types of correlations can be identified in compressed video streams: intra-frame and inter-frame correlations. Intra-frame correlations arise when intra-frame compression (e.g., DCT) is involved *and* the VBR sequence is studied at a smaller level than a frame (e.g., a slice level) [46]. Inter-frame correlations are observed when the VBR stream is studied at the frame level, as in the case of our study.

In several previous studies of video traffic, researchers observed that the *acf* for the frame size sequence of the compressed stream had a negative-exponential shape [52, 23, 24, 46]. A different shape for this function should be expected in MPEG-coded streams because of the existence of three different types of frames in MPEG. These frames are generated *periodically* according to the compression pattern. This results in persistent periodicities in the inter-frame *acf*. The sample *acf* for the frame size sequence of our video stream was computed using the following estimate:

$$\rho(k) = \left(\frac{1}{\sigma^2}\right)\left(\frac{\sum_{i=1}^{N-k}\left[X_i - \overline{X}\right]\left[X_{i+k} - \overline{X}\right]}{N-k}\right) \tag{5.1}$$

where X_i is the size of the ith frame, $N = 41760$, $\overline{X}$ and σ^2 are, respectively, the sample mean and variance. This function is depicted in Figure 5.12.

Although the VBR sequence is being studied at a frame level, the correlation

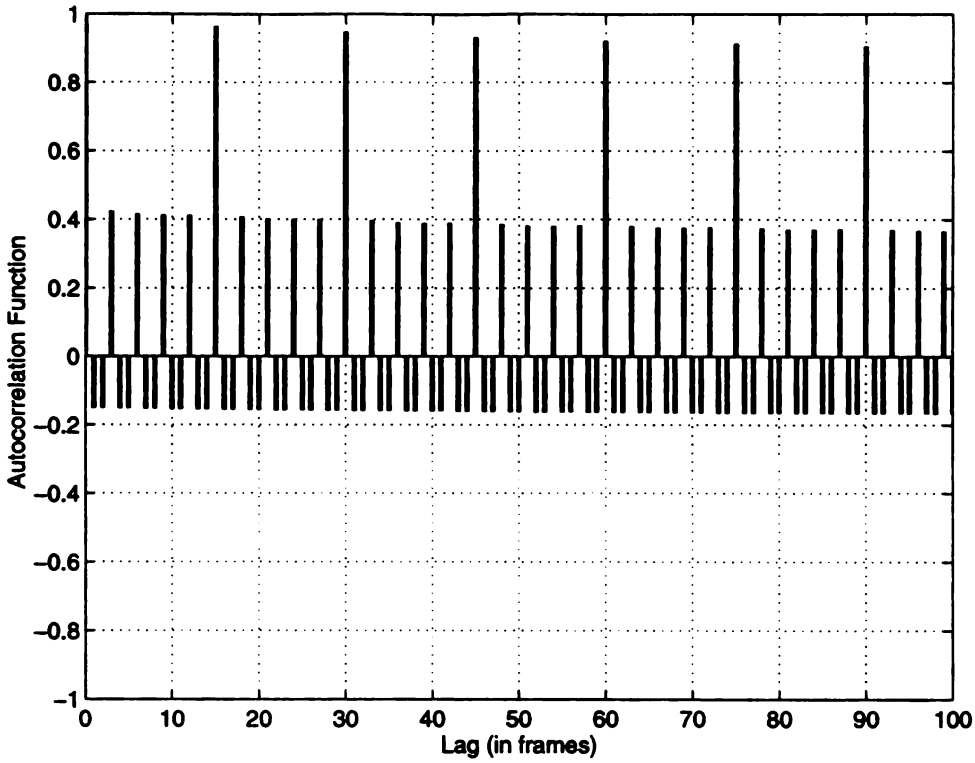

Figure 5.12: Autocorrelation function for the frame size of the VBR sequence.

structure for the sequence is quite complicated and it depicts strong and pseudo-periodic behavior. Two apparent periodic components are observed in Figure 5.12: one at lags of 15 and its multiples, and the other component is at lags of 3 and its multiples. The first component is due to the periodic nature of the compression pattern which is being repeated every 15 frames. The second periodic component results from the correlations between P frames (the distance between two successive P frames within the compression pattern is 3). Both components are bounded by exponentially decaying envelops with large time constants (i.e., slowly decaying). This behavior can be better illustrated by computing the *acf* for the individual I frames (similarly, the P and B frames) as shown in Figures 5.13-5.15.

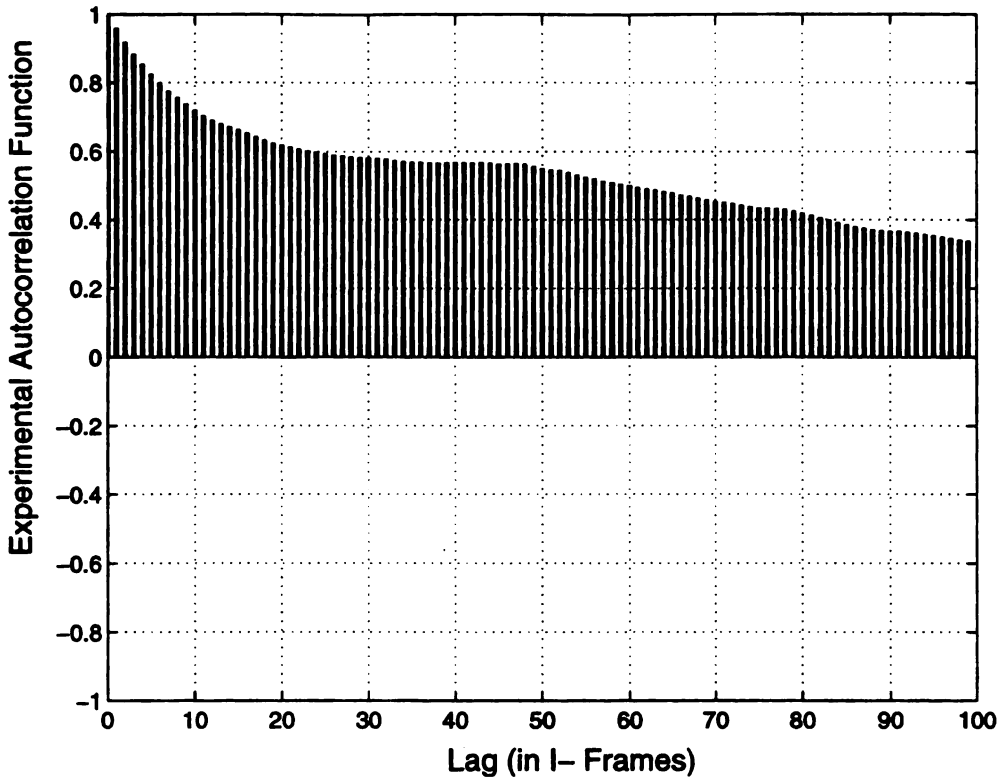

Figure 5.13: Correlations in the *I*-frames subsequence.

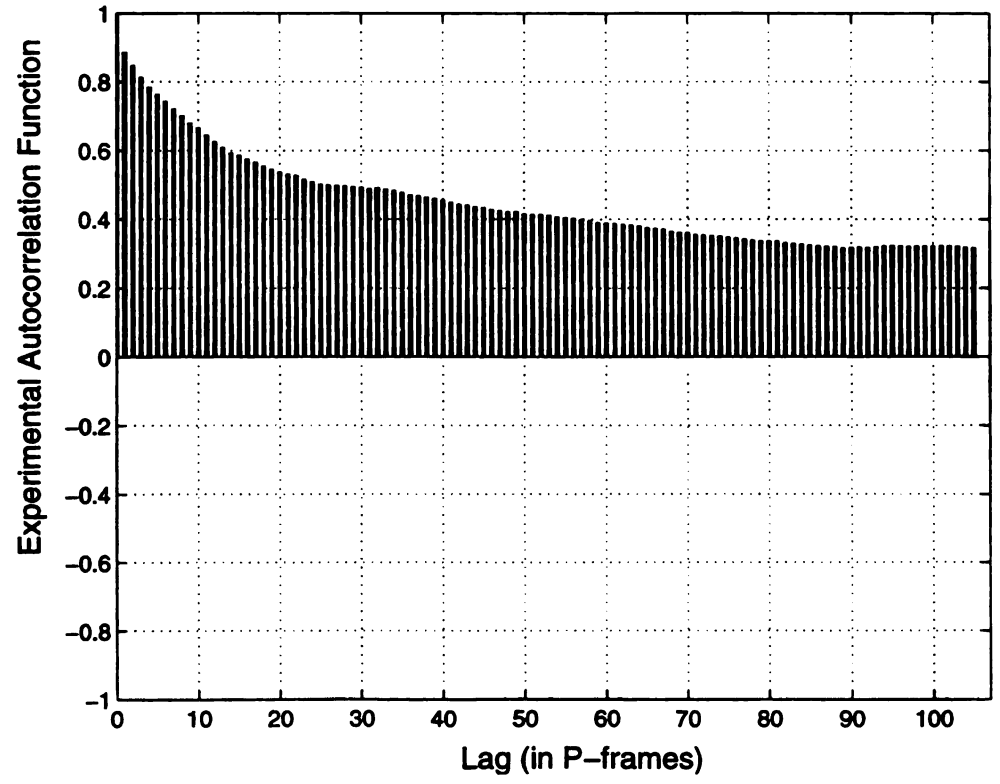

Figure 5.14: Correlations in the *P*-frames subsequence.

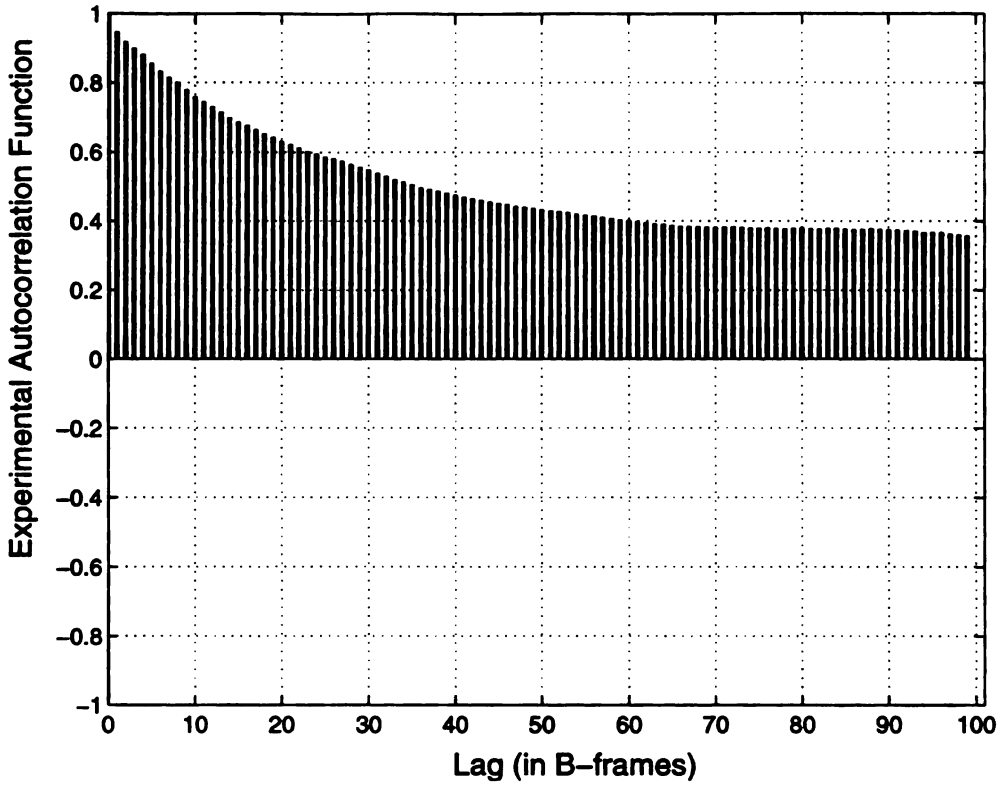

Figure 5.15: Correlations in the *B*-frames subsequence.

5.5 Proposed Traffic Models for MPEG Streams

Typically, a traffic model is used to analyze the queueing behavior of a buffer system. The complexity of compressed video streams precludes such an analytical study. Except for few simple models that were used in the analysis of simple queues (e.g., the fluid model by Maglaris *et. al* [52] for a FIFO infinite queue), models for compressed video have been used to generate synthetic traffic streams for simulation studies. Synthetic streams are needed in situations where empirical data are not available to conduct trace-driven simulations.

The empirical data (i.e., frame size sequence) described in the previous section were used to derive two source models for an MPEG-coded video stream. Since these data belong to a video movie, the proposed models are particularly suitable to characterize the frame size sequence of a compressed video movie. The models can be extended to other types of video (e.g., video teleconferencing) by adjusting their

parameters to match certain statistics of the empirical data. For the purposes of this research, we will only be concerned with video streams from movies.

5.5.1 First Model

5.5.1.1 Model Description

Driven by the need for a simple *parsimonious* traffic model, we propose a model that is based on the compression pattern and the first two moments for the size of individual I, P, and B frames [38, 43]. In this model, a traffic stream consists of a sequence of frames that are generated at a constant rate (e.g., 30 frames per second). Frames are of 3 types: I, P, and B. Frame types are determined according to a given compression pattern (hence, the compression pattern must be known *apriori*). The first frame in a stream can be of any type. Specifically, the type of the first frame is obtained by randomly selecting a location in the compression pattern, and then assigning the frame type in that location to the first frame. Once the type of the first frame (and its location within the compression pattern) is determined, the types and locations of the remaining frames are found easily. For example, in the compression pattern of Figure 5.4, the first frame in a stream can be in any of the 15 locations in the compression pattern (with equal probabilities = 1/15). Suppose that the location of the first frame was randomly decided as the 4th frame in the compression pattern. Then, the first frame is a P frame. The subsequent 11 frames in the stream are of the respective types: *BBPBBPBBPBB*. The compression pattern is then used again, and the next 15 frames have their types exactly as the ones in the compression pattern, and so on.

To model the size of a frame of each type, the three subsequences in Figures 5.9 - 5.11 were used. Recall that each subsequence represents the frame size trace for frames having the same type. The empirical histograms for the three subsequences

are shown in Figures 5.16-5.18 (values in these histograms were normalized so that the total area under the histogram is one). Each histogram was fitted to three continuous *pdf*s: gamma, Weibull, and lognormal, as shown in Figures 5.9 - 5.11. Although the actual frame size is an integer number of cells, we used continuous distributions to simplify the fitting process. This should have a very minor effect on the accuracy of the fitting. When sampling from these distributions, the random samples are truncated to the closest (larger) integer to maintain a whole number of cells in a frame (this is being used in the simulation experiments of an ATM multiplexer for MPEG sources that are generated according to the model).

The true parameters for the candidate distributions are substituted by their *maximum-likelihood estimates* (MLE). These estimates were computed according to the formulae in [4, page 345] for gamma and Weibull densities, and [45, page 337] for a lognormal density.

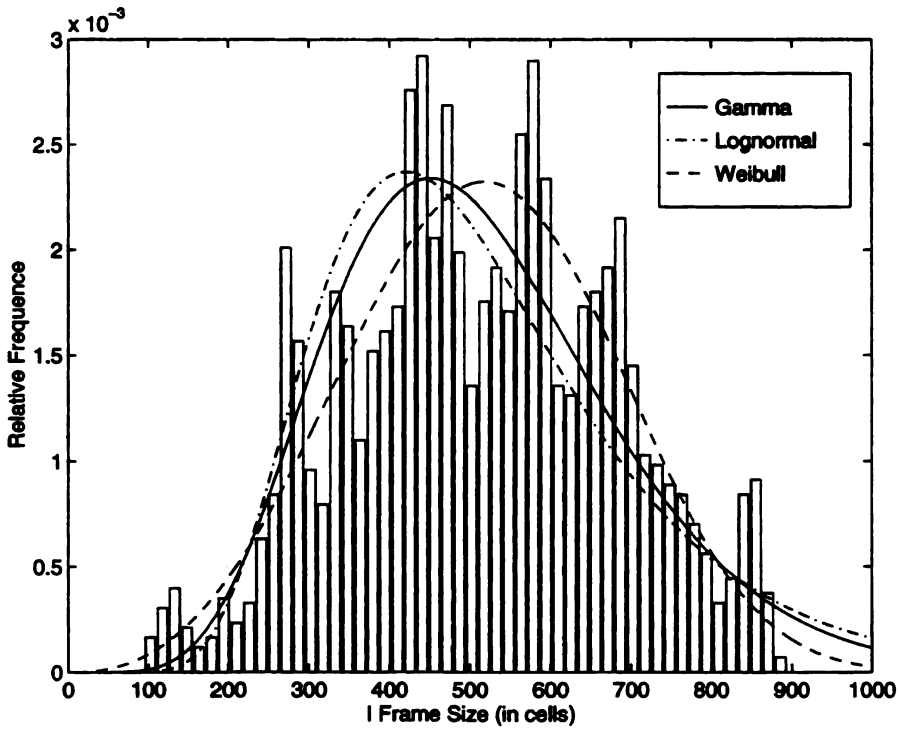

Figure 5.16: Histogram and fitting *pdf*s for the size of an *I* frame.

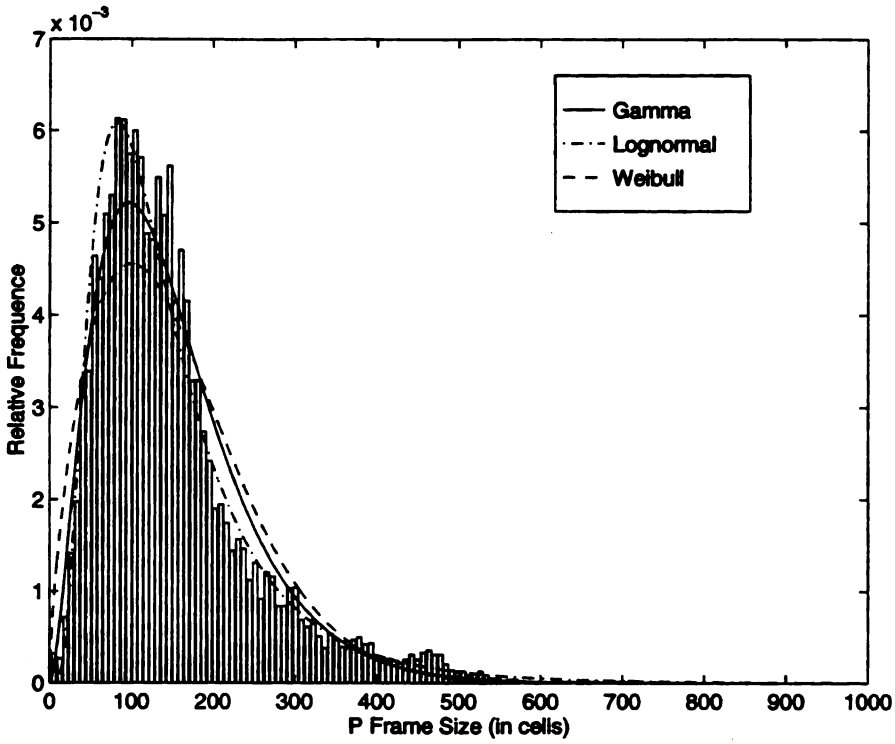

Figure 5.17: Histogram and fitting *pdf*s for the size of a P frame.

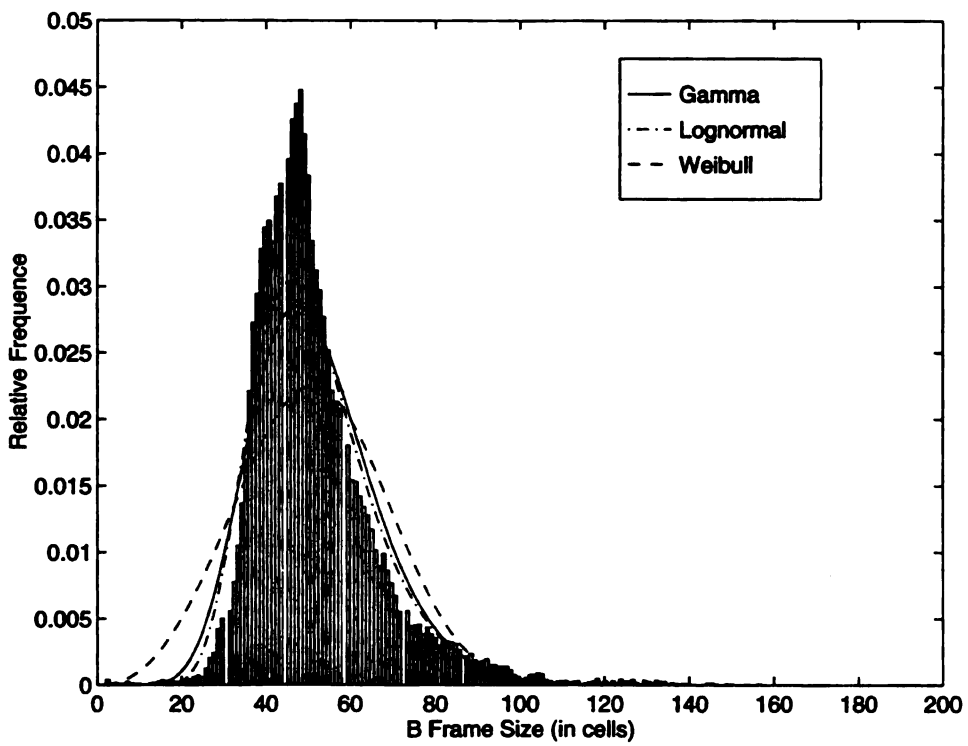

Figure 5.18: Histogram and fitting *pdf*s for the size of a B frame.

5.5.1.2 Frame Size Distribution

From Figure 5.16 it is observed that none of the candidate *pdf*s provides an excellent fit to the frame size histogram for I frames. The multi-mode structure of the empirical histogram requires a more complicated fitting distribution than the examined ones. In order to maintain the simplicity of the model, we restricted our selection process to the three candidate *pdf*s. Any of these functions can be equally used. We chose a lognormal *pdf* as the frame size distribution for I frames. As shown in Figures 5.17 and 5.18, among the three candidate *pdf*s a lognormal *pdf* also provides the best fit to the frame size histograms for P and B frames. The match is best in the case of P frames. For B frames, the empirical distribution is more deterministic than any of the examined functions.

We chose a lognormal *pdf* as the basis of our model. Hence, the number of cells in a frame is assumed to follow a lognormal distribution. The parameters for this distribution vary from one type of frames to another. The lognormal *pdf* for a frame of type T (where $T \in \{I, P, B\}$) has the following form:

$$f_T(x) = \begin{cases} \frac{1}{x\sqrt{2\pi\sigma_T^2}} \exp\left[\frac{-(\ln x - \mu_T)^2}{2\sigma_T^2}\right], & x > 0 \\ 0, & otherwise \end{cases} \tag{5.2}$$

This function f_T is parameterized by a shape parameter $\sigma_T > 0$, and scale parameter $\mu_T \in (-\infty, \infty)$. The mean and variance for a lognormally distributed random variable are, respectively:

$$m_T = \exp\left\{\mu_T + \sigma_T^2/2\right\} \tag{5.3}$$

$$v_T = \exp\left\{2\mu_T + \sigma_T^2\right\}\left(\exp\left\{\sigma_T^2\right\} - 1\right) \tag{5.4}$$

In our model, we substituted the MLE of μ_T and σ_T for their true values. These estimates are given in Table 5.2 for each frame type.

Frame Type (T)	MLE for the lognormal fit	
	$\hat{\mu}_T$	$\hat{\sigma}_T$
I	5.1968	0.2016
P	3.7380	0.5961
B	2.8687	0.2675

Table 5.2: Maximum likelihood estimates for the parameters of the lognormal fits.

In the following, we derive an expression for the frame size distribution for an *arbitrary* frame. Consider an MPEG stream which is generated according to the proposed model. The frame size sequence for this stream is a discrete-time random process $\{X_n : n = 1, 2, \ldots\}$ where X_n is the number of cells in the nth frame. Our objective is to derive the distribution of X_n. Since lognormal *pdf*s are being used in the model, the derived distribution of X_n will be a continuous function, and thus a continuous version of the actual discrete function.

Let S_n be a discrete random variable which describes the location of the nth frame within the compression pattern. The state space for S_n is $\Omega_S = \{1, 2, \ldots, C\}$, where C is the compression period ($C = 15$ for our empirical data). The variable S_n indicates the type of the frame. Based on the compression pattern in Figure 5.4: if $S_n = 1$, then the nth frame is an I frame; if $S_n = 4, 7, 10, 13$, then the nth frame is a P frame; otherwise, the nth frame is a B frame.

According to the model, given $S_n = i$, $X_n \sim$ lognormal distribution with parameters that depend on i. For $i = 1, \ldots, 15$, define $F_i(x) \triangleq \Pr[X_n \leq x / S_n = i]$. Then,

$$F_i(x) = \begin{cases} F_I(x) , & \text{if } i = 1 \\ F_P(x) , & \text{if } i = 4, 7, 10, 13 \\ F_B(x) , & \text{otherwise} \end{cases} \tag{5.5}$$

Notice that $F_i(x)$ does not depend on n since the model assumes that the sizes of same-type frames are *iid*.

It is easy to see that $\{S_n\}$ constitutes a Markov chain with the following transition

probability matrix:

$$\mathcal{P} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & & & \vdots & & & & & & & & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.6}$$

where $\mathcal{P} = [p_{ij}] \triangleq \Pr[S_n = j / S_{n-1} = i]$. The Markov chain $\{S_n\}$ is irreducible and homogeneous. All the states are positive recurrent with mean recurrence time equals to 15. Thus, there is a unique stationary distribution Π such that $\Pi\mathcal{P} = \Pi$. Let $\Pi = [\pi_1\ \pi_2\ \ldots\ \pi_{15}]$. Then, $\pi_i = 1/\text{mean recurrence time} = 1/15$ for all i. The probability distribution function for X_n is given in the following proposition.

Proposition 1 *If $S_1 \sim \Pi$, then the random process $\{X_n\}$ is first-order stationary. In this case, the pdf for X_n is given by:*

$$f(x) = \frac{1}{15}\left(f_I(x) + 4f_P(x) + 10f_B(x)\right) \tag{5.7}$$

Moreover, the mean and variance of X_n are given by:

$$m \triangleq E[X_n] = \frac{1}{15}\left[m_I + 4m_P + 10m_B\right] \tag{5.8}$$

$$v \triangleq E\left[(X_n - m)^2\right] = \frac{1}{15}\left(v_I + m_I^2 + 4(v_P + m_P^2) + 10(v_B + m_B^2)\right) - m^2 \tag{5.9}$$

where the quantities on the right hand sides of (5.7)- (5.9) were given in (5.2), (5.3), and (5.4).

Proof: It is clear that if $S_1 \sim \Pi$, then the Markov chain $\{S_n\}$ is stationary. Let

$F(x,n) \triangleq \Pr[X_n \leq x]$ (for now, we maintain the time dependency). Then,

$$F(x,n) = \sum_{i=1}^{15} \Pr[X_n \leq x,\ S_n = i] \tag{5.10}$$

$$= \sum_{i=1}^{15} \Pr[X_n \leq x / S_n = i] \Pr[S_n = i] \tag{5.11}$$

$$= \frac{1}{15} \sum_{i=1}^{15} \Pr[X_n \leq x / S_n = i] \tag{5.12}$$

$$= \frac{1}{15} (F_I(x) + 4F_P(x) + 10F_B(x)) \tag{5.13}$$

Taking the derivative of $F(x,n)$ with respect to x yields (5.7). Notice that the expression for $F(x,n)$ does not contain n, i.e., $\{X_n\}$ is first-order stationary. From (5.7) it is easy to compute m and v. □

Using the MLE for μ_T and σ_T ($T = I, P, B$) given in Table 5.2, $f(x)$ was computed according to (5.7) and is plotted in Figure 5.19. By comparing Figures 5.8 and 5.19, it is observed that the frame size distribution based on the model is very similar to the actual distribution. This gives strong credibility in the model.

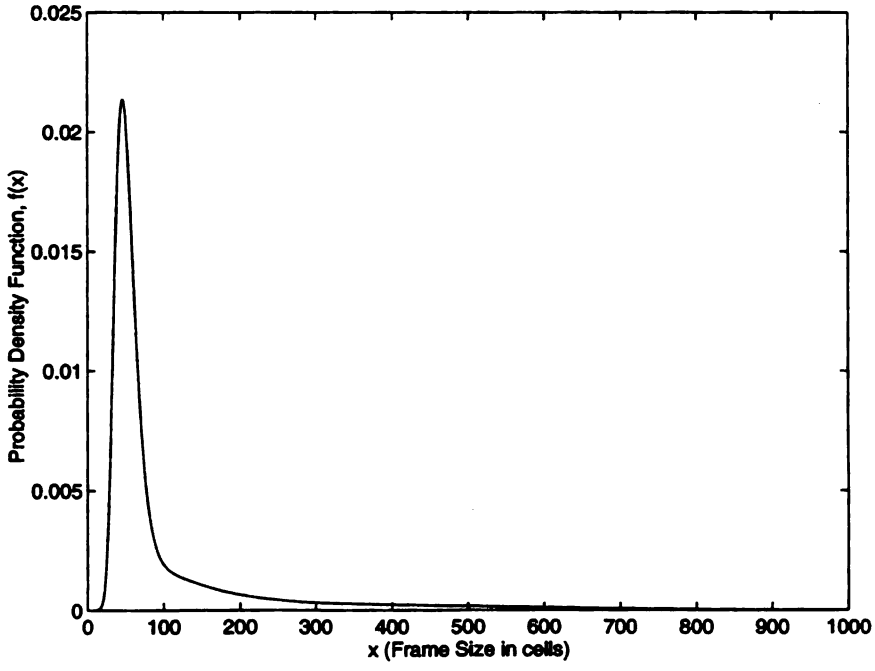

Figure 5.19: The *pdf* for the size of an arbitrary frame based on first model.

5.5.1.3 Autocorrelation Function

When studying the characteristics of compressed video, traffic modelers often try to develop a model in which the *acf* for frame sizes is similar to the empirical *acf*. This is important because of the impact of traffic correlations on the queueing performance. In this section, we derive the *acf* for our model and show its similarity to the empirical one. First, we show that the random process $\{X_n\}$ is second-order stationary (i.e., the joint distribution of the random variables X_n and X_{n+k} does not depend on n). Let $H_{X_nX_{n+k}}(x,y) \triangleq \Pr[X_n \leq x,\ X_{n+k} \leq y]$. In the following proposition, we provide the expression for $H_{X_nX_{n+k}}(x,y)$ and show that this expression does not depend on n. First, define

$$p_{ij}^{(k)} \triangleq \Pr[S_{n+k} = j/S_n = i] = \begin{cases} 1, & \text{if } j = ((i+k-1) \bmod 15) + 1 \\ 0, & \text{otherwise} \end{cases} \tag{5.14}$$

Proposition 2 *If $S_1 \sim \Pi$, then the random process $\{X_n\}$ is second-order stationary, and the joint cumulative distribution function is given by:*

$$H_{X_nX_{n+k}}(x,y) = \frac{1}{15}\left(\sum_{i=1}^{15-k^*} F_i(x)F_{i+k^*}(y) + \sum_{i=15-k^*+1}^{15} F_i(x)F_{i+k^*-15}(y)\right) \tag{5.15}$$

where $k^* = ((i-1) \bmod 15) + 1$, and $F_i(x)$ is given in (5.5) $\forall$ i and x.

Proof:

$$\begin{aligned} H_{X_nX_{n+k}}(x,y) &= \Pr[X_n \leq x,\ X_{n+k} \leq y] \\ &= \sum_{i=1}^{15}\sum_{j=1}^{15} \Pr[X_n \leq x,\ X_{n+k} \leq y,\ S_n = i,\ S_{n+k} = j] \\ &= \sum_{i}\sum_{j} \Pr[X_{n+k} \leq y,\ S_{n+k} = j/X_n \leq x,\ S_n = i] \cdot \Pr[X_n \leq x,\ S_n = i] \end{aligned}$$

But $\Pr[X_{n+k} \leq y,\ S_{n+k} = j/X_n \leq x,\ S_n = i] = \Pr[X_{n+k} \leq y,\ S_{n+k} = j/S_n = i]$

(since S_n contains all the necessary information). Thus,

$$\begin{aligned} H_{X_nX_{n+k}}(x,y) &= \sum_i \sum_j \Pr[X_{n+k} \le y,\ S_{n+k} = j / S_n = i] \cdot \Pr[X_n \le x,\ S_n = i] \\ &= \sum_i \sum_j \Pr[X_{n+k} \le y,\ S_{n+k} = j / S_n = i] \cdot \Pr[X_n \le x / S_n = i]\, \pi_i \\ &= \frac{1}{15} \sum_i \sum_j \Pr[X_{n+k} \le y / S_{n+k} = j,\ S_n = i] \cdot p_{ij}^{(k)} \cdot \Pr[X_n \le x / S_n = i] \\ &= \frac{1}{15} \sum_i \sum_j \Pr[X_{n+k} \le y / S_{n+k} = j] \cdot p_{ij}^{(k)} \cdot \Pr[X_n \le x / S_n = i] \\ &= \frac{1}{15} \sum_i \sum_j F_j(y) F_i(x) p_{ij}^{(k)} \end{aligned} \tag{5.16}$$

which is the same as (5.15). Notice that the expression for $H_{X_nX_{n+k}}(x,y)$ does not contain the index n, i.e., the process $\{X_n\}$ is second-order stationary. □

To derive the *acf* for $\{X_n\}$, let $\rho(k)$ denote the autocorrelation at lag k. Then,

$$\rho(k) \triangleq \frac{C(k)}{v} \tag{5.17}$$

where v is the variance (given in (5.9)) and $C(k)$ is the covariance at lag k

$$C(k) \triangleq \mathrm{E}[(X_{n+k} - m)(X_n - m)] = \mathrm{E}[X_n X_{n+k}] - m^2 \tag{5.18}$$

The following proposition provides the expression for $C(k)$.

Proposition 3 *The covariance at lag k is given by*

$$C(k) = \frac{1}{15} \sum_{i=1}^{15} m_{h(i)} m_i - \left(\frac{1}{15} \sum_{i=1}^{15} m_i \right)^2, \quad \textit{for } k \ge 1 \tag{5.19}$$

where $m_j \triangleq E[X_n / S_n = j]$ *for* $j = 1, \ldots, 15$, *and* $h(i) \triangleq (i + k - 1) \bmod 15 + 1$.

Proof: first notice that $m_j \in \{m_I, m_P, m_B\}$ for $j = 1, \ldots, 15$. Consider

$\mathrm{E}[X_n X_{n+k}] = \mathrm{E}[X_1 X_{1+k}]$ (because of stationarity) for $k \geq 1$.

$$\begin{aligned}
\mathrm{E}[X_1 X_{1+k}] &= \int_0^\infty \Pr[X_1 X_{1+k} > u]\, du \\
&= \int_0^\infty \sum_{i=1}^{15} \Pr[X_1 X_{1+k} > u / S_1 = i] \cdot \Pr[S_1 = i]\, du \\
&= \sum_{i=1}^{15} \pi_i \mathrm{E}[X_1 X_{1+k} / S_1 = i] \\
&= \frac{1}{15} \sum_{i=1}^{15} \mathrm{E}[X_1 X_{1+k} / S_1 = i] \qquad (5.20)
\end{aligned}$$

To determine $\mathrm{E}[X_1 X_{1+k} / S_1 = i]$, observe that the two events $[X_1 / S_1 = i]$ and $[X_{1+k} / S_1 = i]$ are independent (i.e., conditionally independent). Thus,

$$f_{(X_1 X_{1+k}/S_1=i)}(x, y) = f_{(X_1/S_1=i)}(x) f_{(X_{1+k}/S_1=i)}(y) \qquad (5.21)$$

Equation (5.20) becomes:

$$\begin{aligned}
\mathrm{E}[X_1 X_{1+k}] &= \frac{1}{15} \sum_{i=1}^{15} \mathrm{E}[X_1 / S_1 = i]\, \mathrm{E}[X_{1+k} / S_1 = i] \\
&= \frac{1}{15} \sum_{i=1}^{15} m_i m_{h(i)} \qquad (5.22)
\end{aligned}$$

and the assertion is proved. □

Using (5.19) and (5.17), the autocorrelation function for the frame size sequence based on the model was computed and is shown in Figure 5.20.

A comparison between Figures 5.12 and 5.20 reveals that the model reproduces the general shape of the empirical *acf*. This fact gives strong credibility in the model. However, the values for the correlations based on the model are smaller than their empirical counterparts. This should be expected since the model *only* captures correlations induced by the periodicity and deterministic nature of the compression pattern (this type of correlations is not found in compression algorithms other than MPEG). Correlations between frames of the same type are not captured (in the model, the sizes of same-type frames are *iid* with a common lognormal distribution. The parameters of this distribution vary from one frame type to another).

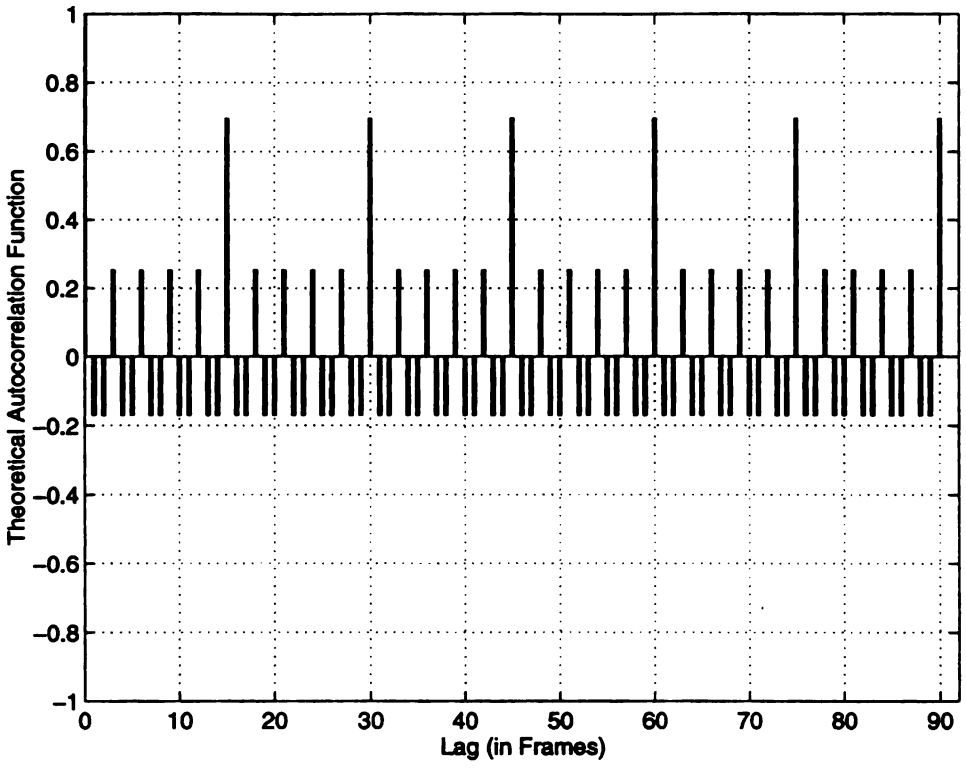

Figure 5.20: Autocorrelation function for the frame size sequence based on first model.

A sample path from the frame size sequence of a synthetic stream is shown in Figure 5.21. Observe that the model does not capture the effect of scene changes that were seen in the empirical trace (Figure 5.5 and 5.6). The empirical histogram and *acf* for the trace of the synthetic stream are shown in Figures 5.22 and 5.23, respectively. Both figures are very similar to the ones derived theoretically.

5.5.2 Second Model

5.5.2.1 Motivation

The model proposed in the previous section captures the frame size histogram as well as the general form of the *acf* of the empirical data. It does not, however, incorporate the scene dynamics. From Figures 5.5 and 5.6, it is observed that the video stream consists of several segments such that the sizes of I frames in each

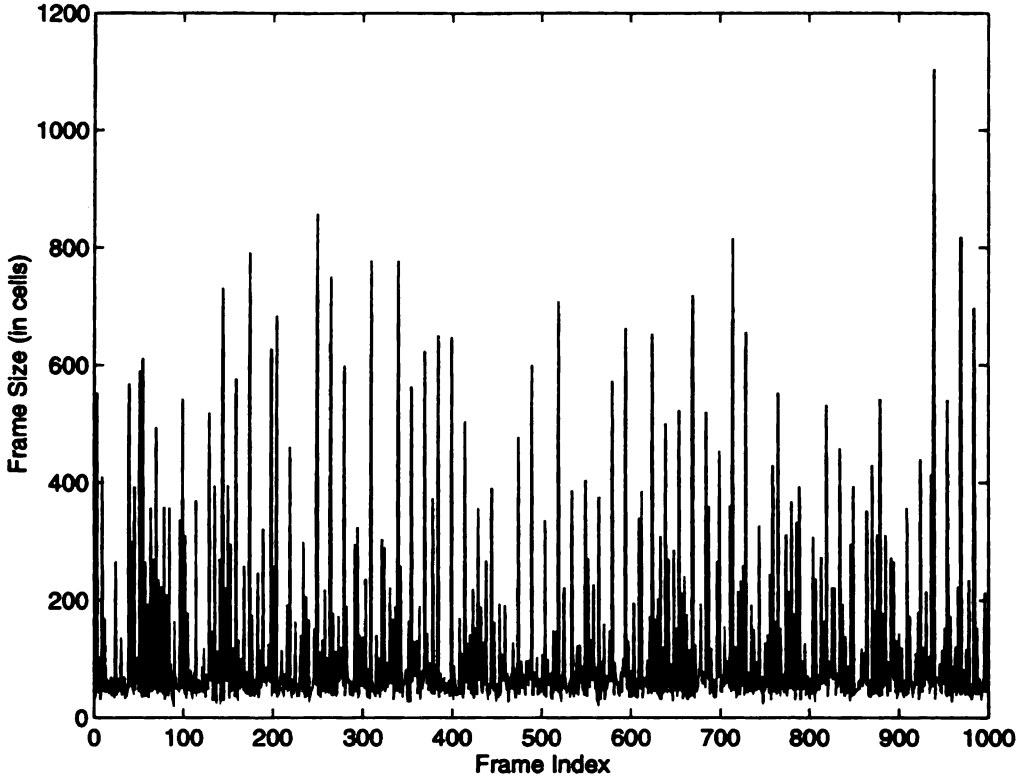

Figure 5.21: Frame size sequence for a synthetic stream based on the first model.

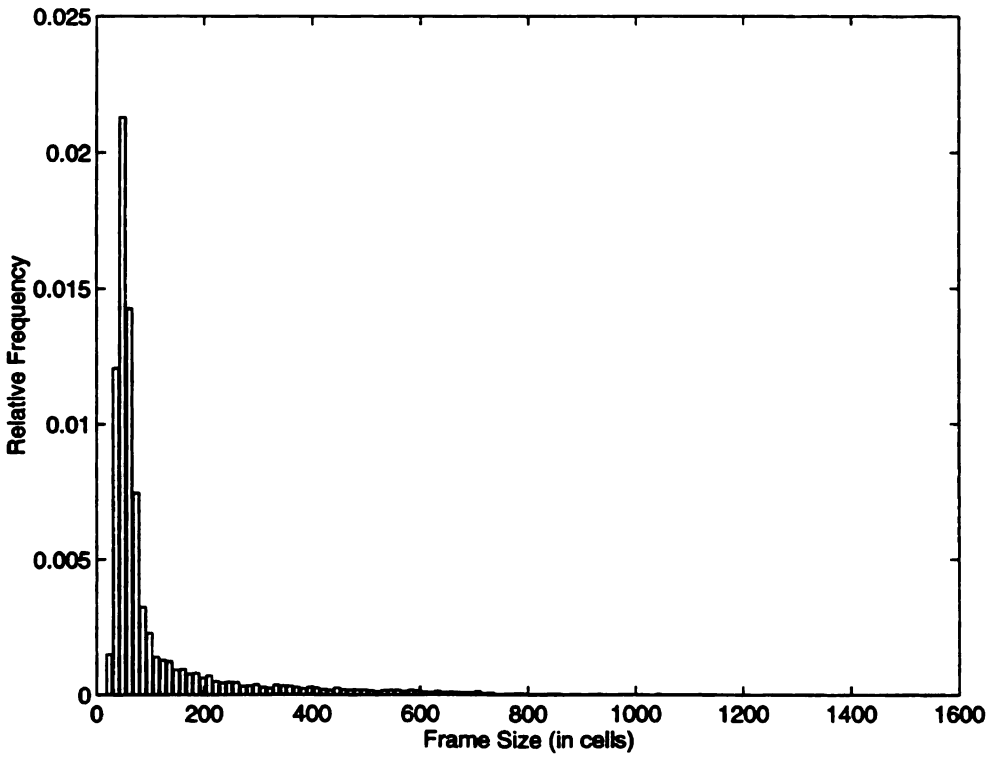

Figure 5.22: Frame size histogram for a synthetic stream based on the first model.

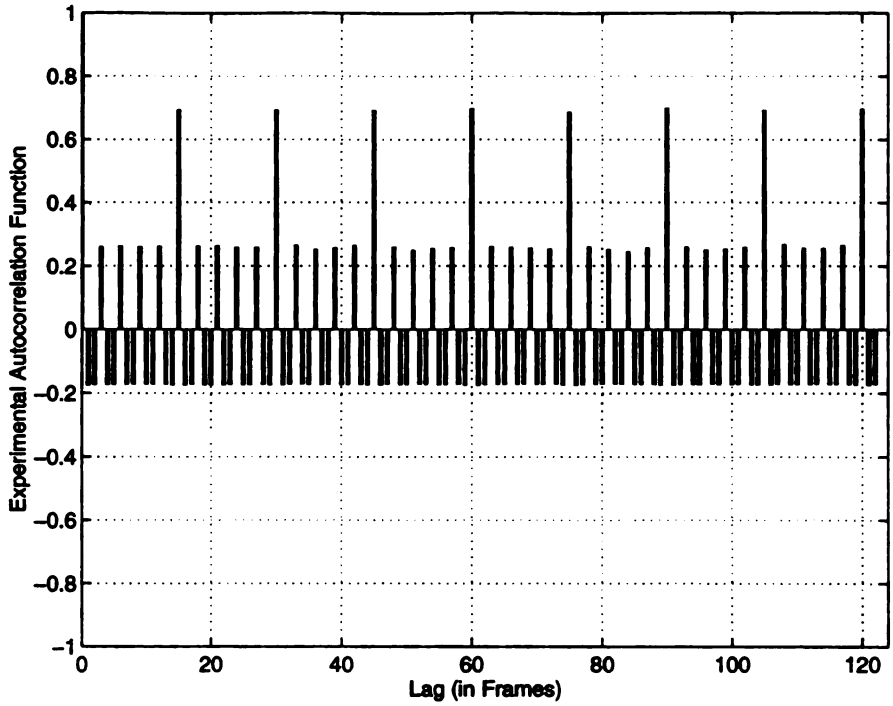

Figure 5.23: Empirical *acf* for the frame size sequence of a synthetic stream generated using first model.

segment are close in value. By watching the movie and simultaneously observing the changes in the frame size sequence, we concluded that each of these segments corresponds to a part of the movie with no *abrupt* view shifts. Each of these segments is being referred to as a *scene*. Note that only an abrupt view shift accounts for the beginning of a new scene. Camera panning and zooming are not considered as scene boundaries. In this section, we develop a second model for an MPEG video stream, which is based on capturing the scene dynamics.

5.5.2.2 Scene Length Distribution

To model the length of a scene (in seconds or in successive frames), one must first obtain a set of empirical data that represent the lengths of actual scenes in the movie. This can be done by watching the movie and measuring the duration of each scene. A disadvantage of this approach is that it requires the availability of the actual movie which must be watched in slow motion to accurately measure scene durations.

Moreover, it is sometimes difficult to visually determine whether a given view shift should be counted as the start of a scene (e.g., when the camera is moving fast but still in a continuous manner). We developed another approach for computing scene durations, which only requires the availability of the VBR trace. This approach, to be presented shortly, was validated by comparing its outcome to the one obtained through visual inspection. For more than 95% of the scenes, both outcomes coincide. In this heuristic approach, we use the fact that a 'sufficient' change in the size of two consecutive I frames is a strong indication of the start of a new scene. Thus, one can use the frame size subsequence of I frames to obtain the scene length data. We assume that the minimum length of a scene is one second (i.e., two consecutive I frames). Let $\{X_I(n) : n = 1, 2, ...\}$ be the I frames subsequence. Suppose that the current scene is the ith scene that started with the kth I frame. The $(n+k+1)$th I frame of the sequence indicates the start of the $(i+1)$th scene if

$$\frac{|X_I(n+k+1) - X_I(n+k)|}{\left(\sum_{j=k}^{n+k} X_I(j)\right)/n} > T_1$$

$$\text{and} \quad \frac{|X_I(n+k+2) - X_I(n+k)|}{\left(\sum_{j=k}^{n+k} X_I(j)\right)/n} > T_2$$

where T_1 and T_2 are two thresholds.

Based on the above approach, the frame size histogram for the scene length (in I frames) is shown in Figure 5.24 using $T_1 = 0.05$ and $T_2 = 0.1$. An exponential fit for this histogram is also shown in the figure. Since the scene length is a discrete quantity (measured in successive I frames), we assume that it follows a geometric distribution (the discrete version of an exponential distribution) with mean = 10.5 I frames. A quantile-quantile plot for the empirical and geometric distributions is shown in Figure 5.25. We also tested the hypothesis of independence between scene lengths based on the runs-up test [45]. At a 95% level of confidence, the hypothesis of independence cannot be rejected.

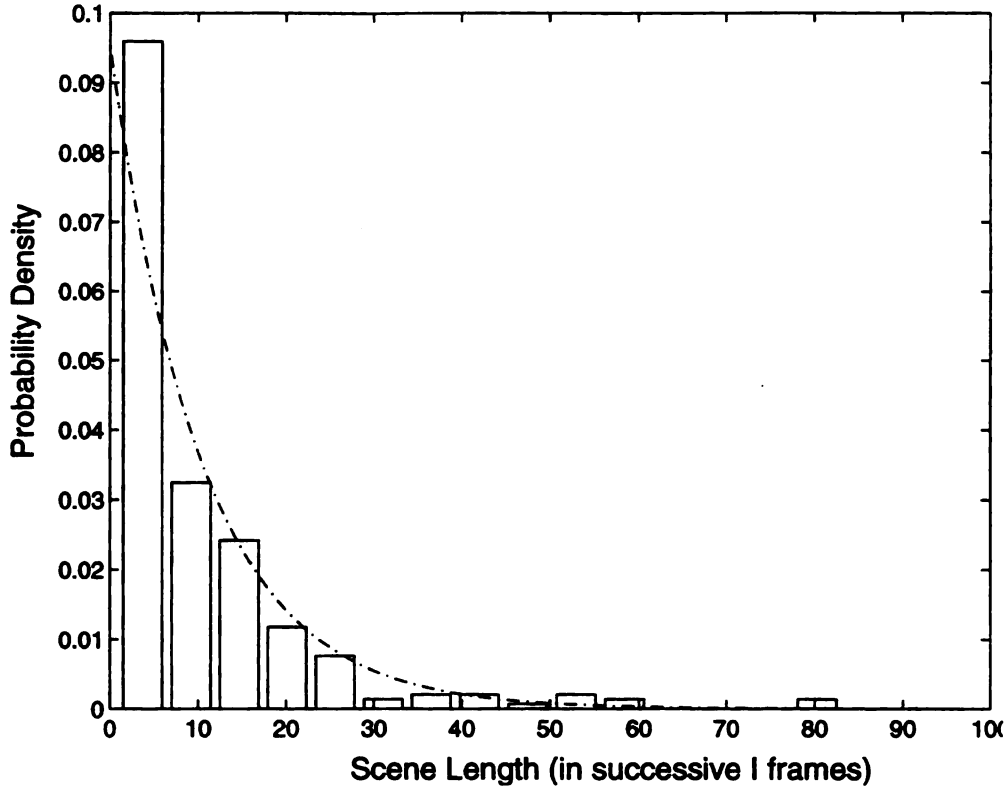

Figure 5.24: Scene length histogram.

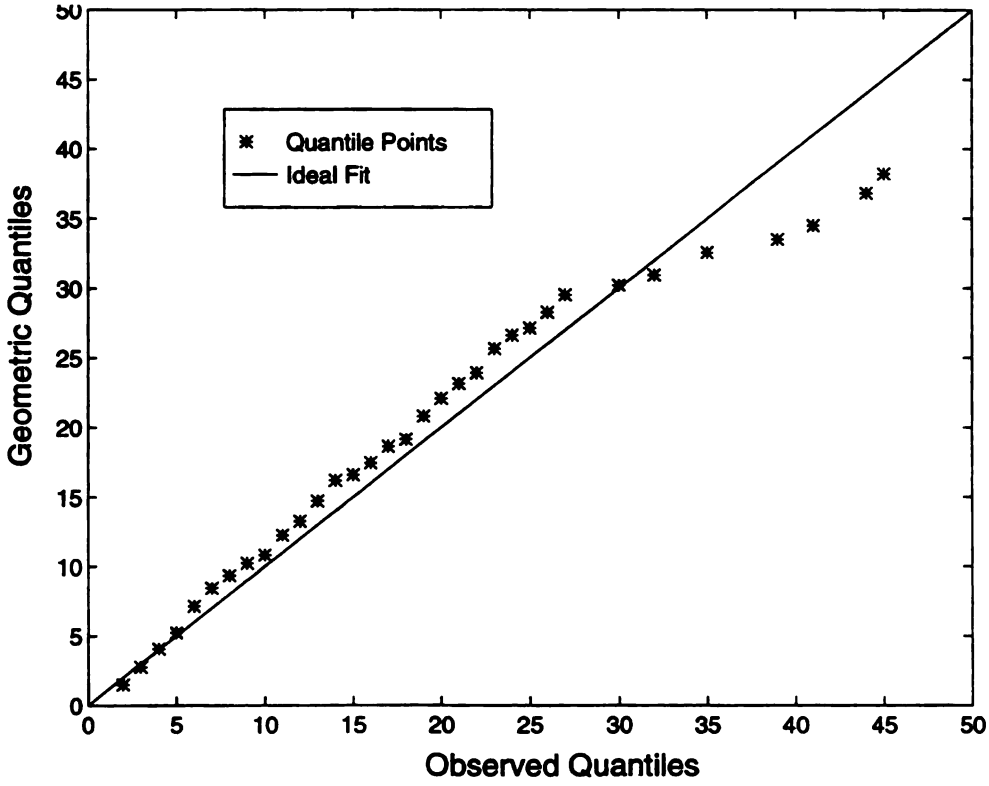

Figure 5.25: Q-Q plot for the scene-length distribution.

5.5.2.3 Frame Size Distribution

Average size of an I frame in a scene.

The model assumes that an MPEG stream consists of several scenes. A scene is being recognized by the overall level of the frame size sequence. We saw in the previous section that this overall level can be identified by the average size of an I frame taken over all I frames within a scene (this average is given in the denominator of the left-hand sides of (5.23) and (5.23)). This average, which we denote by $I_{avg}(j)$ for the jth scene, is itself a random variable that varies from one scene to another. Hence, we are interested in obtaining a suitable approximation to the distribution of I_{avg}. We assume that $\{I_{avg}(j) : j = 1, 2, \ldots\}$ is a sequence of *iid* random variables. Using the empirical values of $I_{avg}(j)$ $(j = 1, 2, \ldots)$, the histogram in Figure 5.26 was constructed. Compared to the frame size histogram of I frames (see Figure 5.16), the frame size histogram of I_{avg} depicts a more regular shape. In Figure 5.26, three *pdf*s are fitted to the histogram (using the MLE for the parameters of these *pdf*s). We adopt a lognormal distribution to characterize the average size of I frames in a scene. The MLE values for the lognormal fit in Figure 5.26 are: $\hat{\mu}_I = 6.0188$ and $\hat{\sigma}_I = 0.4556$.

Sizes of I frames within a scene.

Given the average size of I frames in a scene, we would like to characterize the small variations in the sizes of I frames within that scene. Let $n_I(j)$ be the number of I frames in the jth scene ($n_I(j)$ is geometrically distributed for all j). Denote the size of the kth I in the jth scene by $X_I(k, j)$, where $k = 1, \ldots, n_I(j)$. Let $\Delta_I(k, j) = X_I(k, j) - I_{avg}(j)$ which represents the deviation of the size of the kth I frame in the jth scene from the average size of I frames in that scene. Our objective is to model the sequence $\{\Delta_I(k, j) : k = 1, \ldots, n_I(j)\}$. The difficulty is modeling such a sequence is that for a given j, $n_I(j)$ is very small ($n_I(j) = 10.5$ on the average). To overcome this problem, we assume that the stochastic behavior of $\{\Delta_I(k, j)\}$

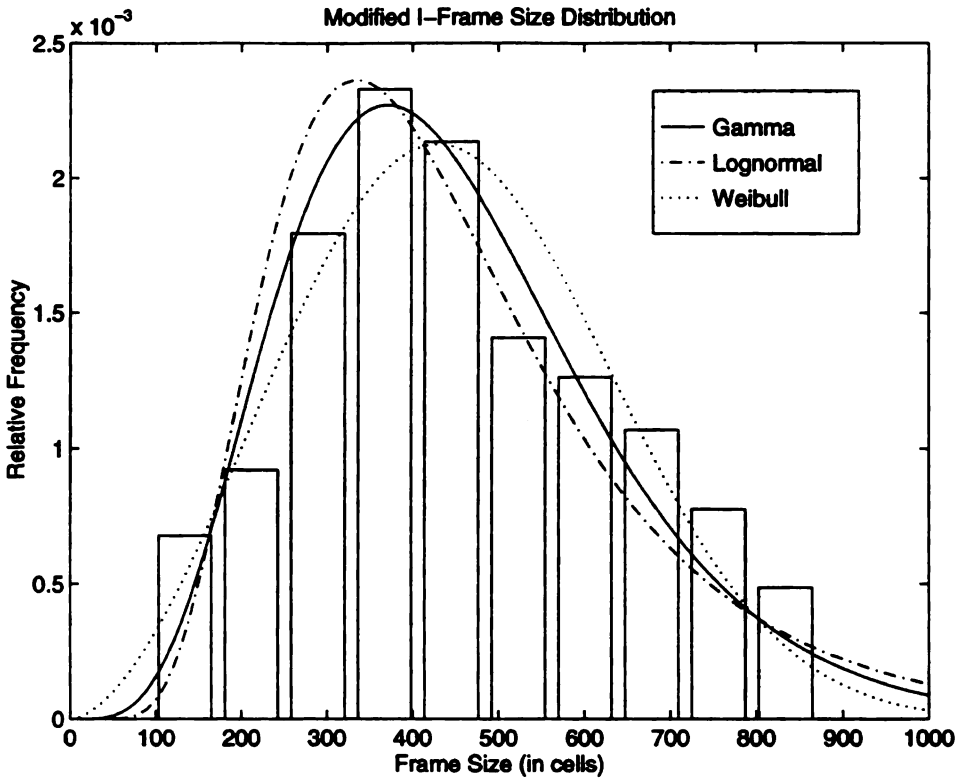

Figure 5.26: Histogram and fitted *pdfs* for the average size of I frames in a scene.

is independent of j, i.e., $\{\Delta_I(k,j)\}$ is invariant with respect to j. Thus, a longer sequence can be constructed by successively concatenating the sequences $\{\Delta_I(k,j)\}$ for $j = 1, 2, \ldots$. This results in the following sequence:

$$\{\Delta_I^*(n) \; : \; n = 1, 2, \ldots\} \triangleq \{\{\Delta_I(k,j) \; : \; k = 1, 2, \ldots, n_I(j)\} \; : \; j = 1, 2, \ldots\} \quad (5.23)$$

It should be expected that the process of concatenation will result in slight errors at the boundaries of the concatenated sequences. This error is of a minor effect on the modeling of $\Delta_I(k,j)$. Intuitively, a time series model is a good candidate to characterize the sequence $\{\Delta_I^*(n)\}$. This sequence is shown in Figure 5.27. Its empirical *acf* is depicted in Figure 5.28 (the bold-font vertical lines).

The shape of the *acf* suggests some form of an autoregressive fit. We believe that an autoregressive of order two (AR(2)) model is appropriate to characterize

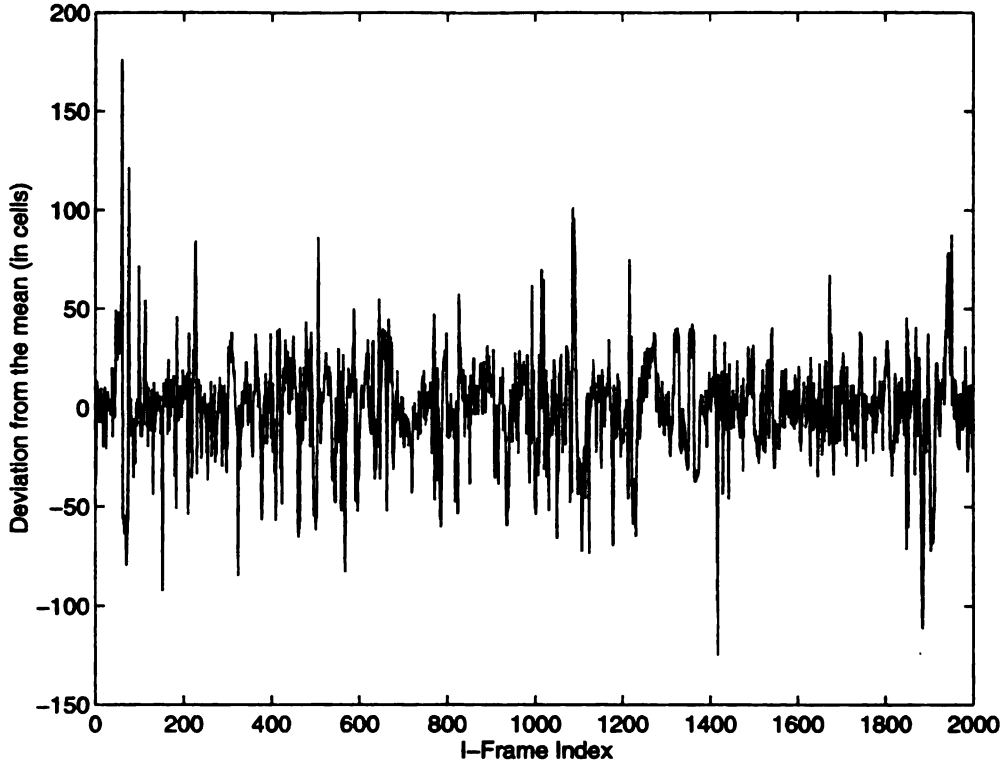

Figure 5.27: The sequence $\{\Delta_I^*(n)\}$.

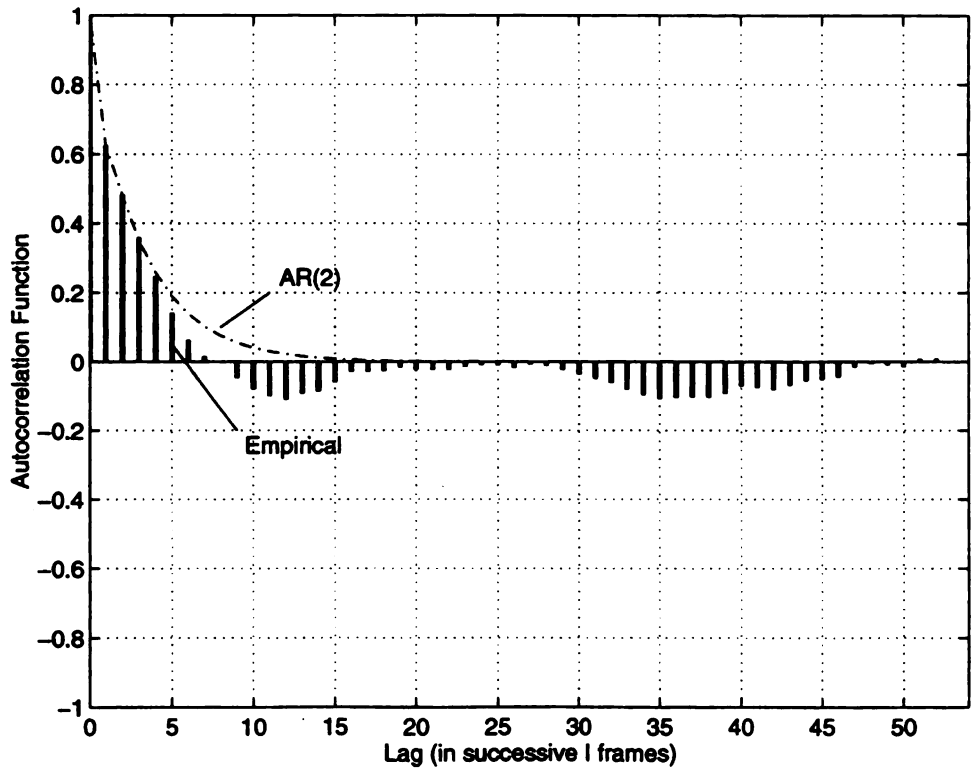

Figure 5.28: The *acf* for the sequence $\{\Delta_I^*(n)\}$ and for an AR(2) model.

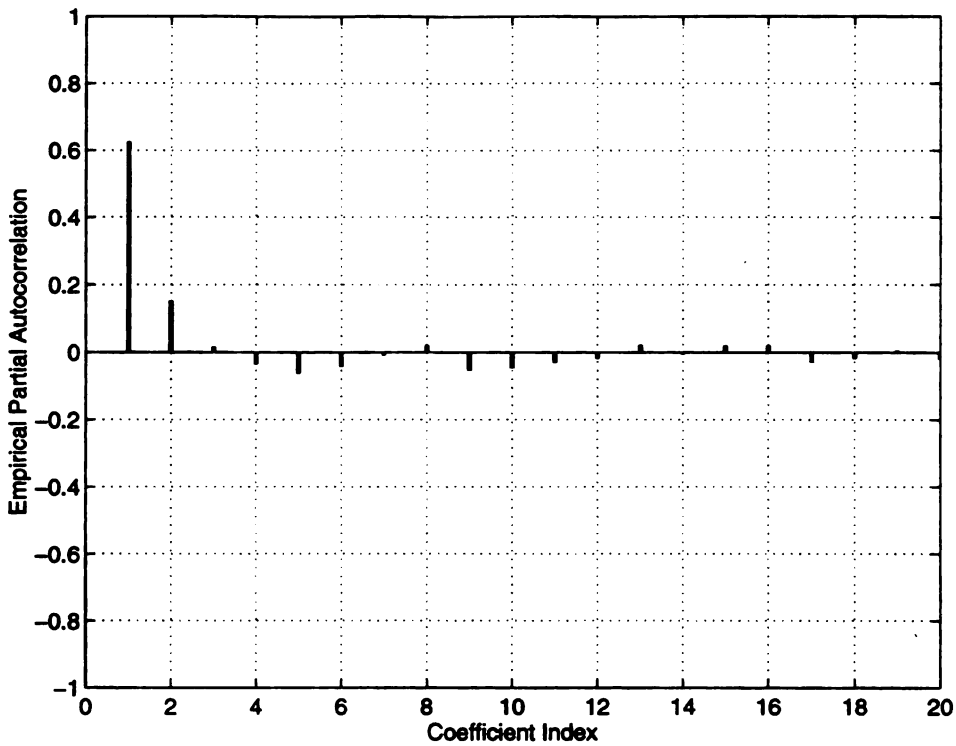

Figure 5.29: Partial autocorrelation function for the sequence $\{\Delta_I^*(n)\}$.

$\{\Delta_I^*(n)\}$. The *acf* for an AR(2) fit is also shown in Figure 5.28. The empirical partial autocorrelation function (*pacf*) was computed and is depicted in Figure 5.29. For a general autoregressive process, the *pacf* provides a means of determining the number of the autoregressive coefficients. It is clear from the *pacf* in Figure 5.29 that the sequence $\{\Delta_I^*(n)\}$ entails two strong autoregressive components (at lag 1 and 2). Thus, the following AR(2) model will be used to characterize $\{\Delta_I^*(n)\}$:

$$\Delta_I^*(n) = a_1\Delta_I^*(n-1) + a_2\Delta_I^*(n-2) + \varepsilon(n) \tag{5.24}$$

where $\{\varepsilon(n)\}$ is a sequence of *iid* random variables whose marginal distribution is often taken as a normal distribution with mean zero and variance σ_ε^2. Since $\Delta_I^*(n)$ has the same marginal distribution as $\varepsilon(n)$, this distribution is found empirically from the histogram of $\Delta_I^*(n)$, which is depicted in Figure 5.30. A normal fit is also depicted (based on MLE parameters), which provides sufficient support to assume a normal

distribution for $\varepsilon(n)$.

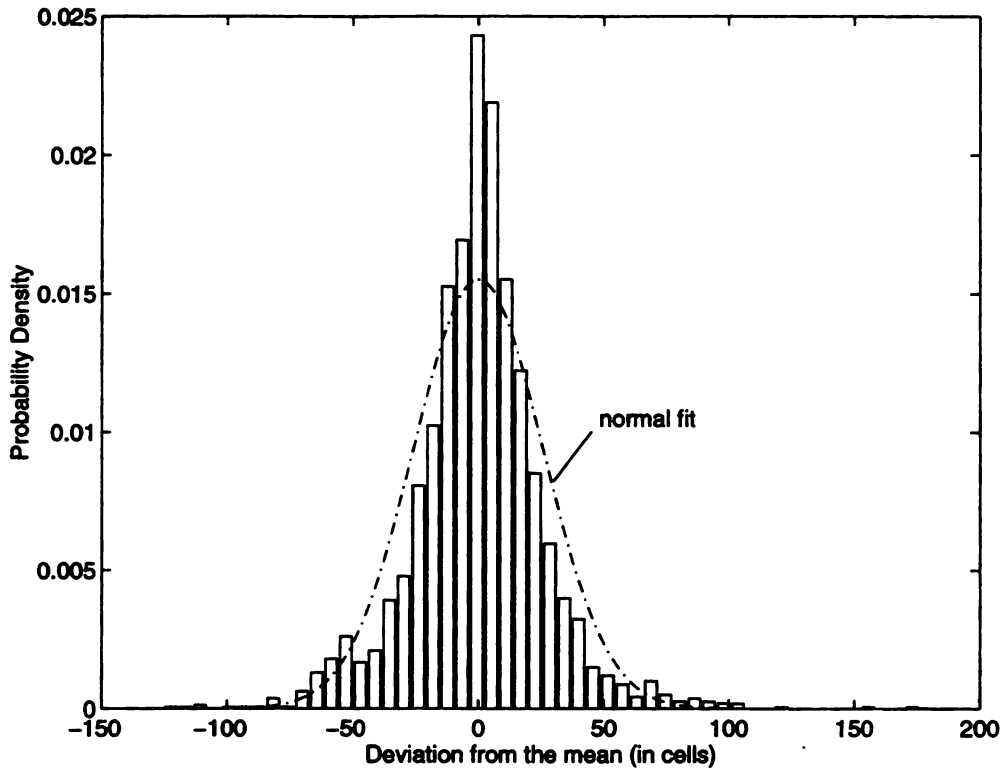

Figure 5.30: Histogram for the sequence $\{\Delta_I^*(n)\}$.

In order to estimate a_1 and a_2 in (5.24), we used Yule-Walker equations [5]:

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} \tag{5.25}$$

where ρ_k is the autocorrelation value for $\{\Delta_I^*(n)\}$ at lag k. Yule-Walker estimates for a_1 and a_2 can be obtained by using the empirical values for ρ_1 and ρ_2 in (5.25). Accordingly,

$$\begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{bmatrix} = \begin{bmatrix} 0.5300 \\ 0.1512 \end{bmatrix} \tag{5.26}$$

The empirical sequence $\{\Delta_I^*(n)\}$ was fitted to an AR(2) model with the above values for a_1 and a_2. A histogram for the residuals is shown in Figure 5.31 along with a normal fit. The *acf* for the residuals (Figure 5.32) clearly confirms the independence

of the residuals (lack of correlation implies independence under the assumption of normality). An estimate for σ_ε^2 can be obtained from the sample variance of the residuals, and was found to be $\hat{\sigma}_\varepsilon^2 = 392.04$.

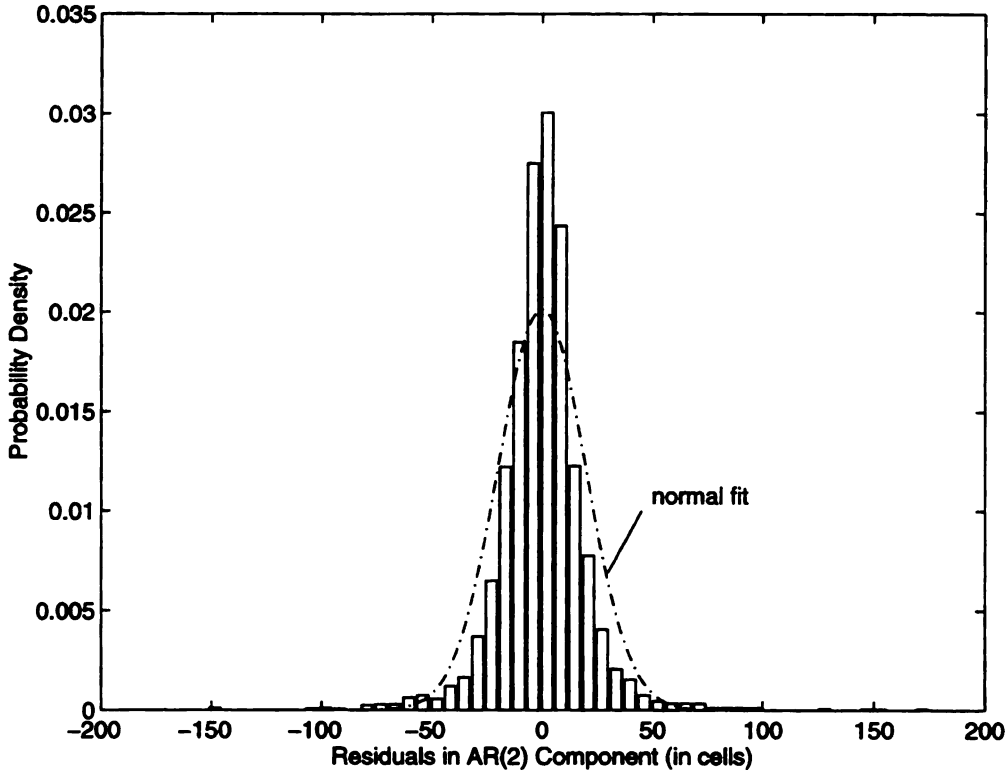

Figure 5.31: Histogram of the residuals in an AR(2) fitted model.

Sizes of P and B frames.

As in the first model, the size of a P or a B frame is assumed to follow a lognormal distribution (with parameters that depend on the type of the frame). The fitting lognormal distributions for both types of frames were shown in Figures 5.17 and 5.18. However, after sampling the size of a P (similarly a B) frame from a lognormal distribution, this size is adjusted to incorporate the overall level of the current scene. For example, if the P frame to be generated is in the jth scene, then the size of this frame is adjusted by multiplying it by the factor $I_{avg}(j)/\overline{I}_{avg}$, where $\overline{I}_{avg}$ is the average of $I_{avg}(j)$ over all j ($\overline{I}_{avg} = 432$ cells for our data). In essence, this multiplying factor reflects the level of the jth scene compared to the overall level of the movie. The

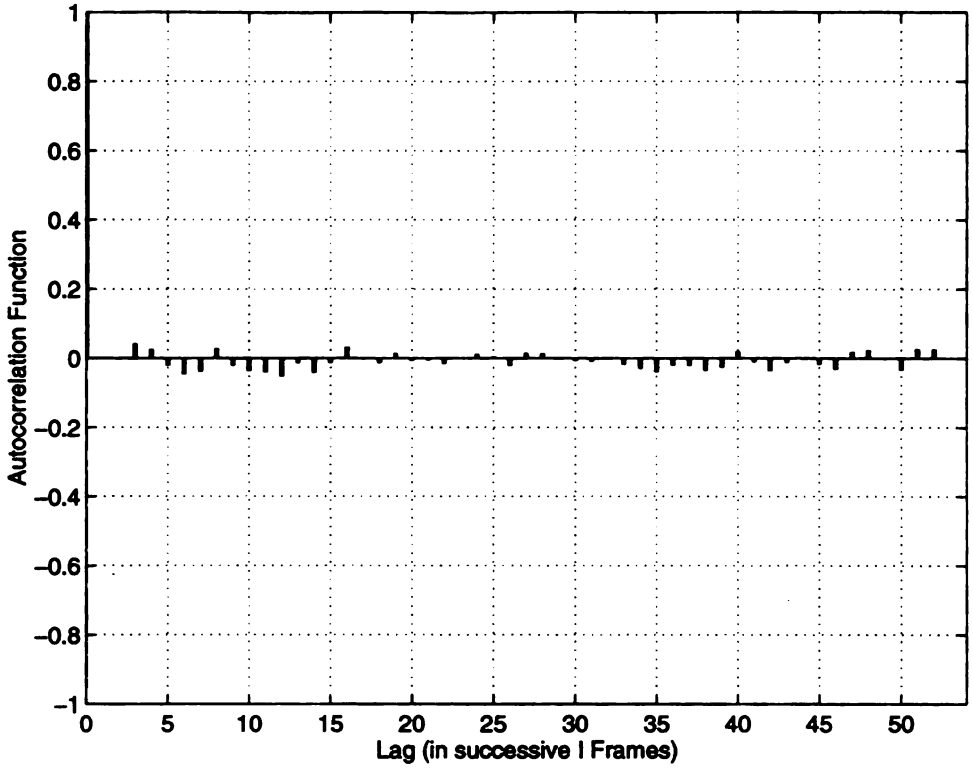

Figure 5.32: Autocorrelation function for the residuals in the AR(2) fit.

same procedure was used with B frames as well.

The algorithm in Figure 5.33 can be used to generate a synthetic MPEG stream based on the present model. Notice that the computer-generated values for the frame sizes are truncated so that they do not exceed their maximum empirical values. In the cases when such maximum values are not known, some upper theoretical bounds could be used instead.

5.6 Queueing Performance under MPEG Streams

In the study of computer networks, traffic models are mainly used to evaluate the queueing performance in the network buffers. Hence, the most important criterion for testing the appropriateness of a given traffic model should be the degree of similarity between the queueing performance under the examined model and that obtain under

Initialization: set $Scene\_max$:= number of scenes
$X \triangleq$ frame size (in cells)
$T \triangleq$ frame type ($T \in \{I, P, B\}$)

Loop:
for $j = 1$ to $Scene\_max$ **do**
compute the length of scene j: $n_I(j) \sim$ Geometric
compute $I_{avg}(j) \sim$ lognormal$(\hat{\mu}_I, \hat{\sigma}_I)$
compute $c \triangleq I_{avg}(j)/\overline{I}_{avg}$
for $i = 1$ to $n_I(j)$ **do**
for $k = 1$ to 15 **do**
$T =$ type of kth frame in the compression pattern
if $T = I$, **then**
$X = I_{avg}(j) + \Delta_I^*(i)$, where
$\Delta_I^*(i) = \hat{a}_1\Delta_I^*(i-1) + \hat{a}_2\Delta_I^*(i-2) + \varepsilon(i)$
end if
if $T = P$, **then**
$X = cY$ where $Y \sim$ lognormal$(\hat{\mu}_P, \hat{\sigma}_P)$
end if
if $T = B$, **then**
$X = cY$ where $Y \sim$ lognormal$(\hat{\mu}_B, \hat{\sigma}_B)$
end if
end for
end for
end for

Figure 5.33: Generating a synthetic stream based on the second model.

the actual traffic. Other criteria such as the similarity in the *acf* structure and the marginal distribution are still useful in the preliminary stages of model development. However, in many cases the queueing performance under a traffic model may be noticeably different from the queueing performance of a real source, despite a relatively similar first and second order statistical properties.

To verify the appropriateness of our models, we report in this section the results from simulation experiments of an ATM multiplexer for MPEG streams. In addition to testing the models, the simulation results are used to study the actual (i.e.,

trace-driven) multiplexing performance for MPEG streams (this is important since we have no previous reference to the actual queueing performance under MPEG traffic streams). The multiplexer (Figure 5.34) is modeled as a finite capacity queueing system with buffer size B (in cells) and one server with service rate C. A FIFO service discipline is assumed. The input to the multiplexer consists of N MPEG-coded video streams. Two types of simulation experiments were conducted. The first type is trace-driven (i.e., using the actual VBR trace). The second type of experiments was based on synthetic streams generated according to the proposed models. In either case, a video source consists of 41760 frames. Cells in each frame are evenly distributed. All generated streams were based on the compression pattern of Figure 5.4. First, we consider the performance using the empirical frame size trace.

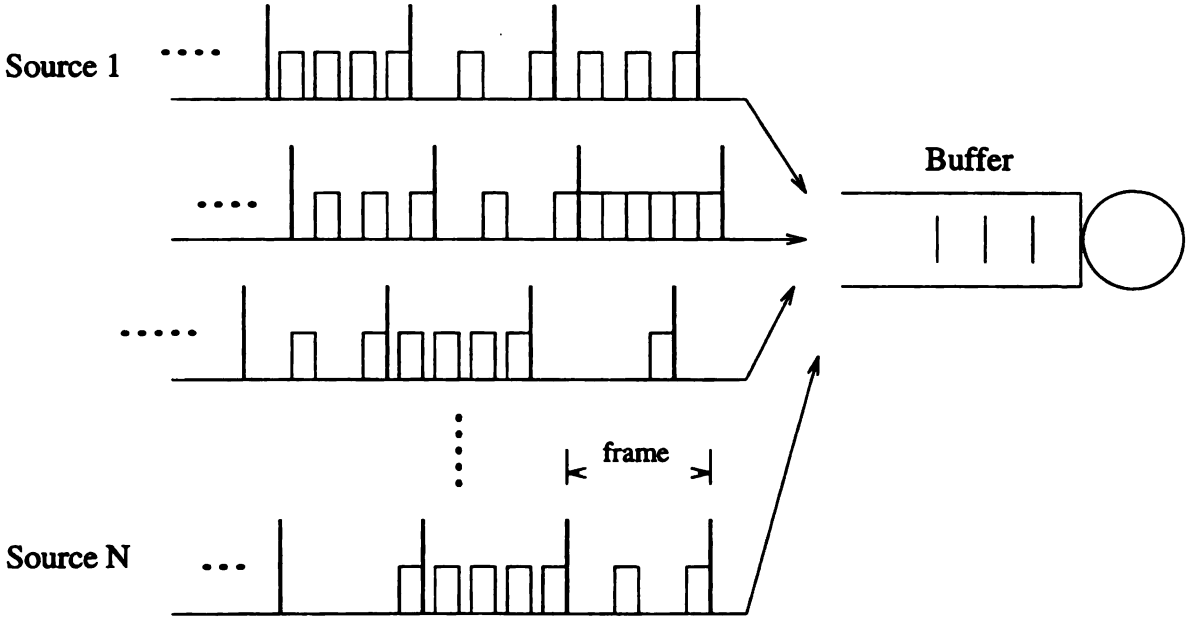

Figure 5.34: A multiplexer for MPEG streams.

5.6.1 Trace-Driven Simulations

In the following experiments, traffic streams based on the empirical trace were used to drive simulations. Since only one trace is available, an approach due to Heyman *et al.*

[24] was used to generate multiple streams from the empirical trace. In this approach, the video sequence is assumed to be homogeneous in some sense. The VBR trace is arranged as a circular list by connecting its ends. To generate a traffic stream, the first frame in that stream is randomly selected from the circular list. The remaining 41759 frames are selected sequentially following the first frame. This process is similar to a random shift with rotation in sequential lists. The procedure is repeated for each multiplexed stream. A drawback of this method is that it ignores possible cross-correlations among different streams (these correlations result from using one trace to obtain all the streams). For our purposes, this method was used only to study the multiplexing of MPEG streams, and not to validate the traffic models. To reduce the effect of synchronization among sources, the starting times for the video streams were selected randomly from a uniform *pdf* (over a frame period). The service rate for the multiplexer was adjusted to obtain a desired level of utilization U which is defined by the ratio of the aggregate arrival rate from all multiplexed sources to the service rate.

Figure 5.35 depicts the average cell loss rate, P_{avg}, versus the buffer size, B, using different numbers of multiplexed sources: $N = 1$ (no multiplexing) and $N = 10$. The results are shown using two levels of utilization: $U = 60\%$ and $U = 80\%$. Observe that P_{avg} for $N = 1$ is surprisingly large under relatively light load ($U = 60\%$) and large buffer size ($B = 600$). This is true despite the fact that each stream is smoothed by spacing cells in each frame. Another important observation is the improvement in the cell loss performance when video streams are multiplexed. For $N = 10$, P_{avg} is the average cell loss rate of the ten sources. Not all sources experience this rate. The maximum (P_{max}) and minimum (P_{min}) loss rates among the ten sources are shown in Table 5.3, for $U = 60\%$ and $U = 80\%$ and several values of B.

The average cell loss rate versus the link utilization is shown in Figure 5.36 using different N and $B = 150$. It is evident that the loss rate drops significantly when N

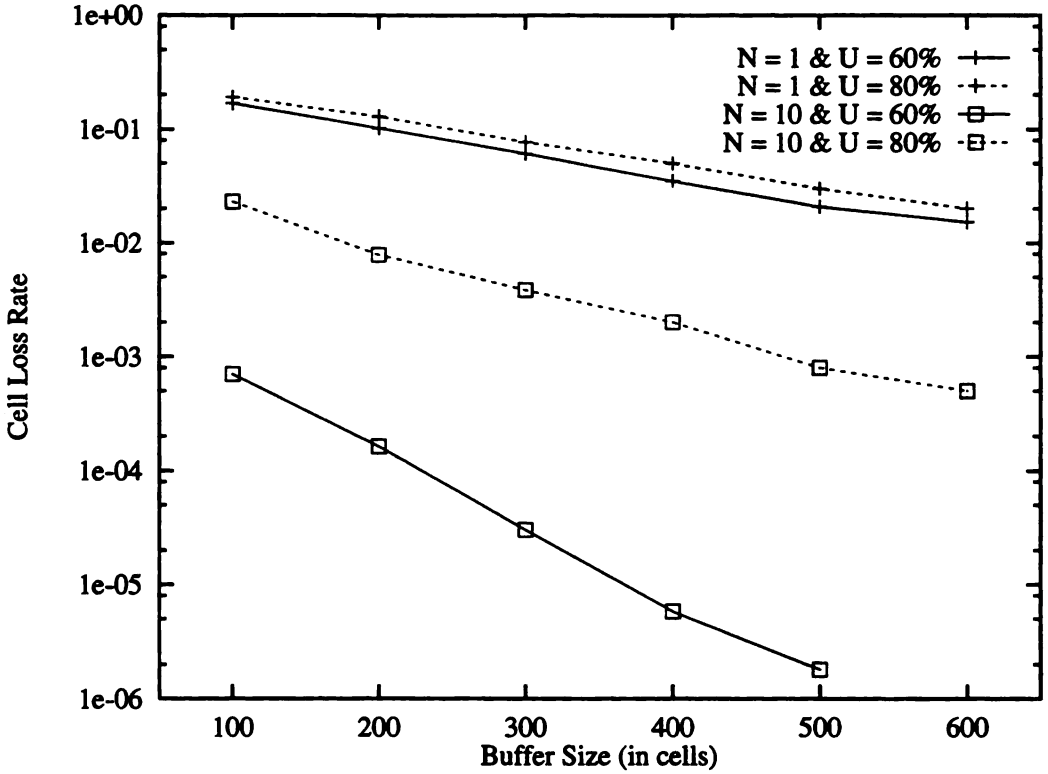

Figure 5.35: Average cell loss rate versus buffer size in trace-driven simulations.

Buffer Size	$U = 60\%$			$U = 80\%$		
	P_{avg}	P_{max}	P_{min}	P_{avg}	P_{max}	P_{min}
100	6.99E−4	1.05E−3	3.98E−4	2.29E−2	5.64E−2	7.04E−3
200	1.62E−4	3.14E−4	8.81E−5	7.83E−3	1.66E−2	4.04E−3
300	3.04E−5	6.30E−5	1.63E−5	3.85E−3	6.86E−3	1.24E−3
400	5.80E−6	1.30E−5	2.20E−6	2.05E−3	4.18E−3	8.59E−4
500	1.80E−6	5.05E−6	2.20E−7*	8.32E−4	2.60E−3	6.05E−4
600	0*	0*	0*	5.47E−4	9.11E−3	2.62E−4

(* loss rates are less accurate due to small or zero number of lost cells)

Table 5.3: Average, maximum, and minimum cell loss rates.

increases at a fixed U. Note that for certain values of U and N no cell losses were observed.

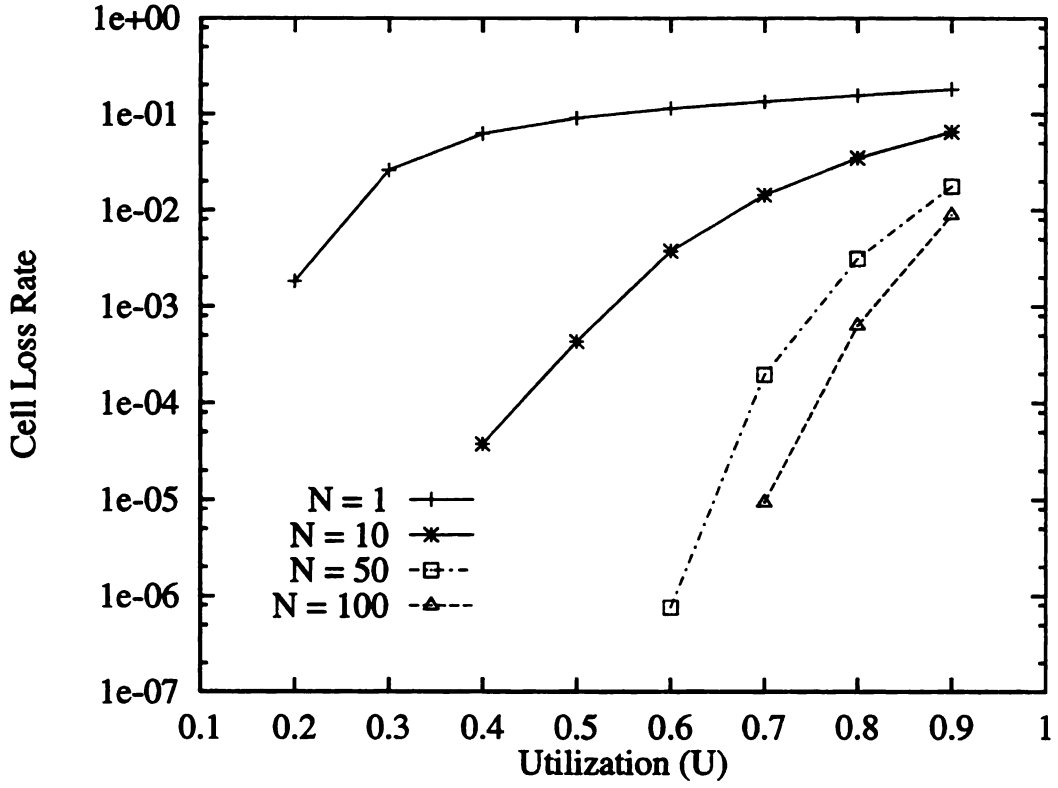

Figure 5.36: Average cell loss rate versus utilization.

The figure clearly indicates that a considerable gain can be obtained by multiplexing VBR video streams. We define the following quantity which will be referred to as the *multiplexing gain*:

$$G_p(n) \triangleq \frac{n \times \text{Utilization at } N = 1 \text{ and } P_{avg} = p}{\text{Utilization at } N = n \text{ and } P_{avg} = p} \tag{5.27}$$

For a given P_{avg} and N, the utilization can be obtained graphically from Figure 5.36. Accordingly, $G_p(n)$ was found for various n and for $P_{avg} = 10^{-2}$ and $P_{avg} = 10^{-3}$ as shown in Table 5.4.

n	$G_p(n)$	
	$p = 10^{-2}$	$p = 10^{-3}$
1	1	1
10	4.1	3.4
100	31.1	21.4

Table 5.4: Multiplexing gain at different loss rates.

5.6.2 Queueing Performance Based on Proposed Models

In this section, we evaluate the appropriateness of the two proposed traffic models from the queueing performance perspective. A FIFO queue with finite capacity B is assumed. Input traffic consists of only one stream. Three cases are considered. In the first case, the stream is the empirical trace. In the two other cases, the stream is generated according to the first and second traffic models, respectively. In each of the following experiments, a traffic stream consists of 41760 frames, generated at the rate of 30 frames/sec. The service rate is adjusted to obtain a given level of utilization. When the traffic stream is generated based on one of the two models, each simulation experiment is repeated 5 times with different random sequence. The 90 percent confidence interval are computed based on the five independent runs. When the empirical trace is used, only a single run is used in each experiment (no random numbers are used in this case).

The cell loss rate versus buffer size is shown in Figure 5.37 using two levels of utilization: $U = 40\%$ and $U = 60\%$. The short vertical lines along the the cell loss curves for the first and second models indicate the 90% confidence intervals based on five replications (using the t distribution with 4 degrees of freedom). For a small buffer size, the performance based on either traffic model is very close to that based on the empirical trace. This is true for both levels of utilization. Intuitively, the difference in queueing behavior becomes smaller as the traffic intensity is increased (at $U = 80\%$ the queueing behavior for the trace and the model-based streams were almost identical). However, as the size of the buffer increases, results from the two

models start to deviate from the results based on the empirical trace. At $U = 40\%$, the trace-based cell loss rate decreases more rapidly than the loss rate according to any of the models. The first model seems slightly more accurate than the second model in terms of reproducing the cell loss rate of the empirical trace. Notice that the second model has relatively large confidence intervals, meaning that it would require a larger number of independent replications to obtain reliable results than would be required in the first model.

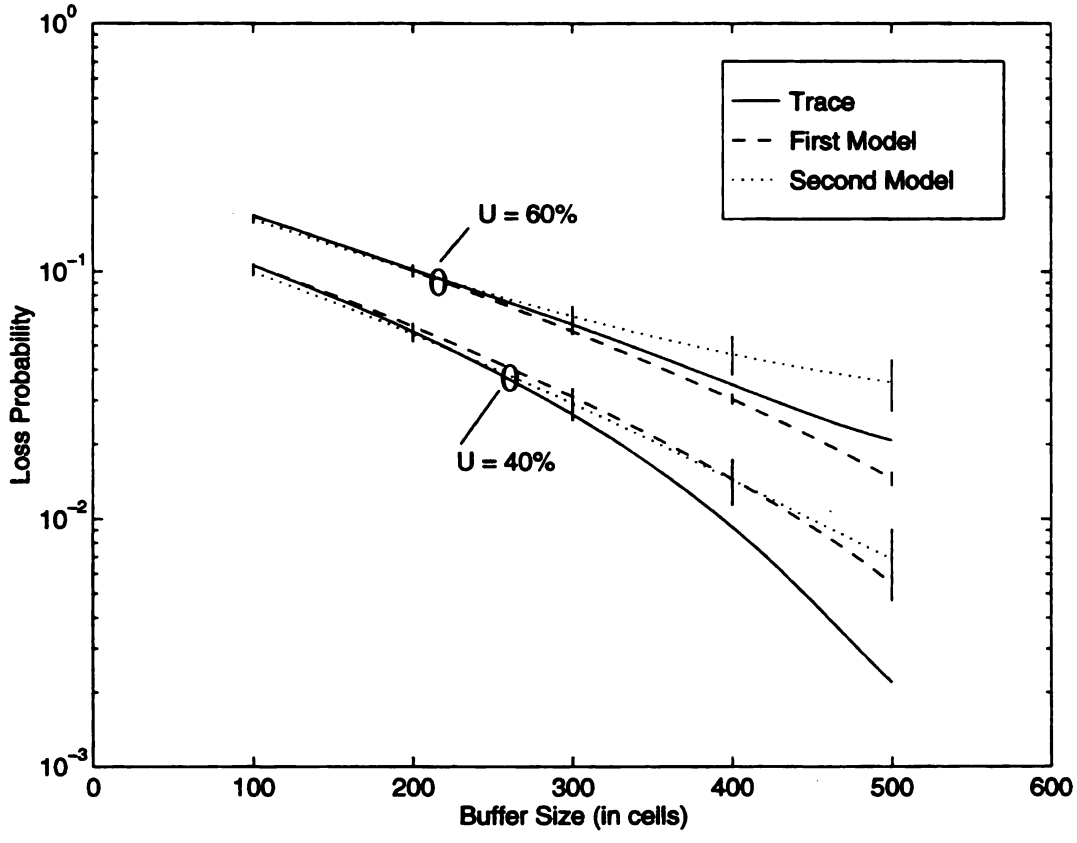

Figure 5.37: Average cell loss rate versus buffer size.

Figure 5.38 depicts the relationship between the average queue length q and the buffer size (B). When $U = 40\%$, the increase in q as a function of B for a model-based stream (using either model) is similar to that of the trace. When $U = 60\%$, the second model provides closer results to trace-based results than the second model.

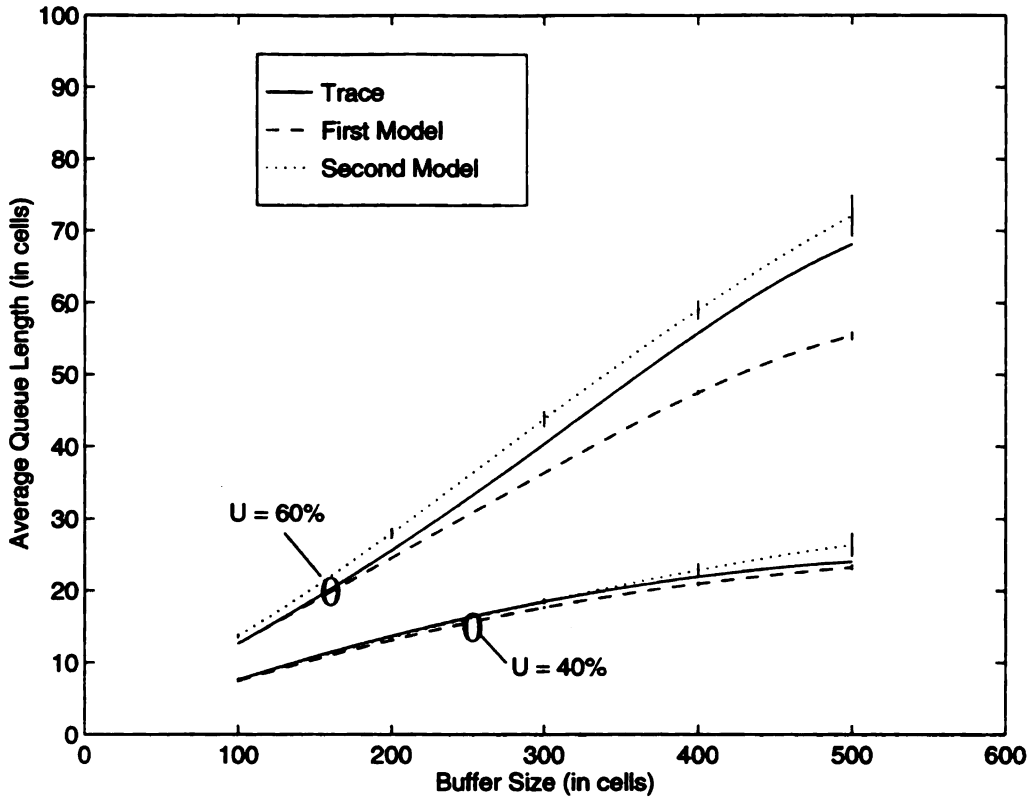

Figure 5.38: Mean queue length versus buffer size.

In all cases, we observed that the queueing results based on the second model upper bounded the trace-based results. Between the two models, we decided to use the second model to characterize the video traffic component in the study of NTCD/MB. Our choice is justified by the need to provide statistical performance guarantees in NTCD/MB and the fact that the second model produces conservative performance results compared to the actual (i.e., trace-based) results. Hence, guaranteeing the QoS requirements of traffic streams generated by the second model will consequently guarantee the QoS requirements of the actual video streams.

Chapter 6

The Performance of NTCD/MB under Multimedia Traffic

6.1 Introduction

In this chapter, we use the various results obtained in previous chapters to demonstrate the performance of the NTCD/MB mechanism under multimedia traffic. For our purposes, "multimedia traffic" refers to the combination of several independent types of traffic, including data, voice, and video streams. Therefore, a multimedia source in which voice and video bits, for example, are intermixed within a single stream is not included in the current study, mainly due to the lack of data regarding the format of such a source. Nevertheless, the applicability of NTCD/MB extends as well to this type of streams.

The main goal of this chapter is to describe how buffers and bandwidth resources under NTCD/MB can be efficiently allocated to various traffic streams, while, simultaneously, guaranteeing the sets of QoS requirements associated with these streams. In theory, this is a constrained optimization problem whose exact analytical solution is not tractable. This is due to the difficulty in analyzing the queueing behavior un-

der video traffic streams that are generated according to the models developed in the previous chapter. Even for a simple FCFS (First Come First Serve) queue, analytical results under video traffic models are not tractable. In the case of data and voice streams, analytical queueing results can be obtained using the fluid approach, as described in *Chapter 4*. Consequently, we used both analytic and simulation approaches to study the queueing performance of the NTCD/MB mechanism under data, voice, and video streams. In addition, we provide near-optimal allocation of bandwidth and buffer resources for such streams given that their QoS requirements are satisfied. Although particular sets of QoS requirements will be assumed throughout the study, the same treatment can be similarly applied to any given sets of QoS requirements.

6.2 Input Traffic and Associated QoS Requirements

We assume that the overall traffic is composed of three types: data, voice, and video. Each type consists of several independent streams that are generated according to the models below. A set of QoS requirements is associated with each type of traffic. The input streams arrive at a buffer system that implements an NTCD/MB mechanism. The number and sizes of the buffers used in NTCD/MB are generally determined according to the QoS requirements of the traffic. Streams that are admitted to the same buffer are statistically multiplexed. In the following, subscripts 1, 2, and 3 will be used, respectively, to refer to quantities associated with data, voice, and video traffic types.

Data traffic consists of N_1 data sources. Each source is characterized by alternating ON/OFF periods. A fluid representation is used to model a data source. It is assumed that the durations of ON periods (similarly, OFF periods) constitute a sequence of *iid* random variables which are exponentially distributed. To provide a realistic

description of a data source, we assume that the mean duration for an OFF period is four times the mean duration of an ON period (i.e., $\tau_{OFF}^{(1)} = 4\tau_{OFF}^{(1)}$). On the average, a data source transmits 200 ATM cells during an ON period (this corresponds to slightly more than 10K bytes). These cells arrive at a constant rate $\lambda_1 = 100$ Kbits/sec. Data streams require the following cell loss rate to be guaranteed (with no requirements on cell delay):

QoS requirements for data traffic: cell loss rate $\epsilon_1 \leq 10^{-3}$

Voice traffic consists of N_2 streams which, similar to data streams, have the ON/OFF behavior. Again, each voice stream is modeled using the fluid approximation. In this case, however, the mean duration of an OFF period is chosen to be about 1.85 times the mean duration of an ON period. This choice is made in compliance with previous measurements taken for adaptive differential pulse code modulation (ADPCM) voice sources with silent detectors [22]. In these measurements, it was found that the mean durations of ON and OFF periods in a voice stream are, respectively, 352 msec and 650 msec. As will be shown in the next section, the ratio of the OFF to ON mean durations plays a significant role in determining the appropriate buffer size needed to achieve a given cell loss rate. During ON periods, voice cells arrive at a rate of $\lambda_2 = 32$ Kbits/sec (which is the standard rate for an ADPCM voice stream). Since voice traffic is delay-sensitive, it has a delay requirement. We assume that voice cells have a maximum queueing delay requirement in addition to a cell loss requirement:

QoS requirements for voice traffic:	cell loss rate $\epsilon_2 \leq 10^{-4}$
	maximum queueing delay $d_2 \leq D_2$

where two values for D_2 will be used: 352 msec and 35.2 msec.

Video traffic consists of N_3 streams. A stream is represented by the second traffic model that we developed in *Chapter 5*. Recall that in this model assumes three frame

types that are generated in a deterministic fashion based on the compression pattern of Figure 5.4. Frames are generated at a rate of 30 frames/sec. Cells within a frame are evenly distributed. The algorithm in Figure 5.33 was used to generate each video stream. To reduce the effect of synchronization among different streams, the type of the first frame in each stream is obtained by randomly selecting the location of the first frame from the 15 locations in the compression pattern, and then assigning the type of the frame in the selected location to the first frame. Thereafter, the types of the remaining frames are taken sequentially according to the compression pattern. In addition, we assume that the starting instant for each stream is random with a continuous uniform distribution over a period of 1/30 second.

With respect to their role in reconstructing the video signal at the receiver, MPEG frames vary in their relative significance depending on the type of the frame. Specifically, I frames are more important than P and B frames since I frames are used as reference points to P and B frames. Thus, if an I frame (or part of it) is lost, all P and B frames between this I frame and the next I frame cannot be properly decoded at the receiver. To protect I frames from excessive cell losses, two classes of loss priority are assigned to video cells, with cells in I frames having higher priority over cells in P and B frames [37]. An NTCD mechanism is required to realize the two classes of priority. Notice that in our study priority classes are assigned on a frame basis. A different approach to priority assignment in an MPEG stream was used in [58] where priorities were assigned on a macroblock basis, with possibly high and low priority cells in every frame. Clearly, this latter approach does not take into consideration the different levels of importance associated with different types of frames. We assume the following QoS requirements for video traffic:

QoS requirements for video traffic:

- cell loss rate for I cells $\epsilon_3^{(1)} \leq 10^{-4}$
- cell loss rate for P and B cells $\epsilon_3^{(2)} \leq 10^{-2}$
- maximum queueing delay $d_3 \leq D_3$

Two values for D_3 were used: 33 msec and 10 msec.

In order to guarantee the aforementioned QoS requirements for data, voice, and video traffic streams, an NTCD/MB mechanism of the form shown in Figure 6.1 is needed. Following the general design methodology that was outlined in the last section of *Chapter 2*, three buffers would be used to support three levels of delay performance. Each buffer accommodates cells of one of the three traffic types. The total service rate (i.e., link capacity) is divided among the three buffers such that each buffer is guaranteed a fraction of the total capacity. These fractional capacities are the quantities C_1, C_2, and C_3 in Figure 6.1, with $C = C_1 + C_2 + C_3$. The service discipline is work-conserving, so that if one buffer is empty at some time, the residual capacity is given to the following buffer (in a cyclic fashion). Effectively, the perceived service rate in each buffer can be higher than the guaranteed service rate (which is being indicated by the "$\geq$" signs in the figure). Our major concern for the remaining sections of this chapter is: (1) determine the appropriate values for the fractional capacities, (2) dimension the sizes of the three buffers, and (3) appropriately partition the space in Buffer 3, such that the QoS requirements for the three types of traffic are satisfied.

6.3 Efficient Allocation of Buffers and Bandwidth

The buffer system shown in Figure 6.1 contains dependencies among the contents of the three buffers. For a given set of QoS requirements, finding the optimal sizes of the buffers and the right amounts of bandwidth to be allocated to each buffer can be analytically done under the fluid assumption. This was shown in *Chapter 4* for two buffers, and can be extended in a straightforward manner to any number of buffers, under the assumption that each traffic stream is characterized by a fluid model. However, for the type of video being considered here, the fluid representation

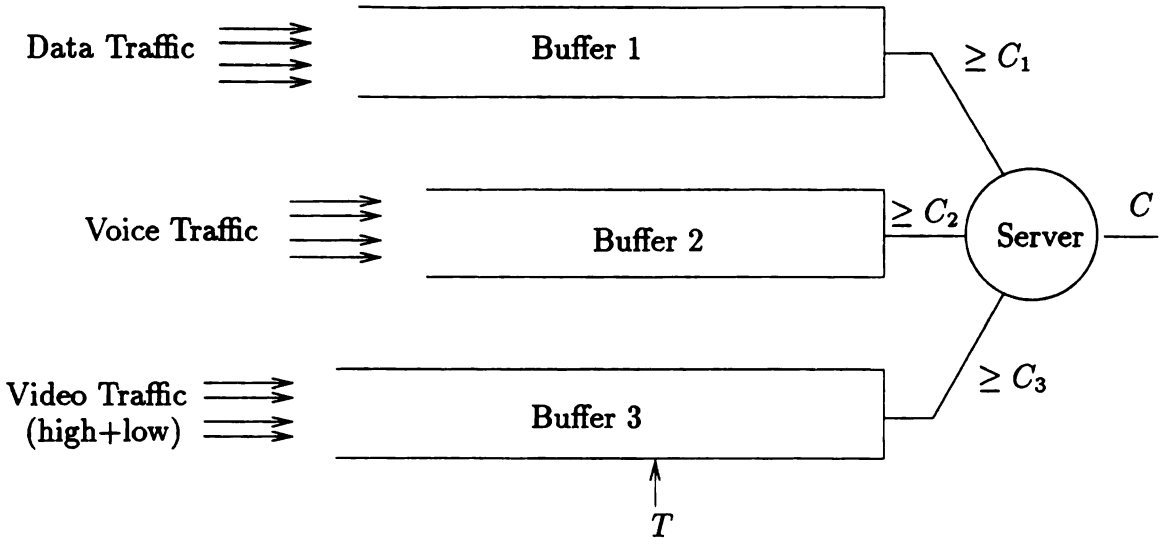

Figure 6.1: Structure of NTCD/MB for data, voice, and video streams.

is not appropriate. The two models which were proposed in *Chapter 5* can accurately characterize MPEG streams. Unfortunately, queueing analysis under such models is not tractable.

In order to dimension the underlying system (using the second video model in characterizing MPEG video streams), we resorted to a two-stage approach. In the first stage, we considered a non-work-conserving service discipline. This allowed us to consider each buffer independent of the other two buffers. We then optimized the size and allocated bandwidth for each buffer (given the QoS requirements associated with input traffic to that buffer). This was done analytically for Buffers 1 and 2 (using fluid analysis), and via simulations for Buffer 3. In the second stage, we used simulations to study the performance of the complete system assuming a work-conserving discipline, and using the buffers sizes and bandwidth computed from the first stage. Intuitively, the performance observed in the second stage is expected to be better than the desired QoS requirements (which were guaranteed under a non-work-conserving assumption). Depending on the difference in performance between the first stage and second stages, we further reduced the size of buffers or, equivalently, the amount of bandwidth so

that the observed performance approaches the desired QoS requirements. The first stage is described next.

6.3.1 Buffer-Bandwidth Tradeoff for Data Traffic

Consider Buffer 1 which accommodates cells from N_1 data streams. The total bandwidth capacity allocated to these streams is C_1. The offered load is given by:

$$U_1 = \frac{\left(\frac{\tau_{ON}^{(1)}}{\tau_{ON}^{(1)}+\tau_{OFF}^{(1)}}\right)\lambda_1 N_1}{C_1} \tag{6.1}$$

It is clear that U_1 is inversely proportional to the amount of bandwidth per source (C_1/N_1). For a fixed number of multiplexed streams, the cell loss rate depends on the buffer size and the allocated bandwidth per source. Hence, for a given loss rate, a tradeoff exists between the size of the buffer and the bandwidth allocated for each source. The cell loss rate for data traffic can be estimated analytically from the complementary buffer distribution computed using the fluid approach (see *Chapter 4*). This is shown in Figures 6.2 and 6.3 for $N_1 = 1$ and $N_1 = 10$, respectively.

For a given N_1, the cell loss probability can be reduced by increasing the buffer size (B_1) or by decreasing the offered load (i.e., increasing C_1/N_1). When U_1 is reduced below a certain level, say U_1^*, the amount of bandwidth that is allocated for each source exceeds the peak rate of the source. In this case, buffering becomes unnecessary. The value of U_1^* is given by:

$$U_1^* = \frac{\tau_{ON}^{(1)}}{\tau_{ON}^{(1)} + \tau_{OFF}^{(1)}} = 20\% \tag{6.2}$$

Figures 6.2 and 6.3 can be used to compute the possible sets of (buffer size, offered load) pairs that result in loss rate of 10^{-3}; the QoS requirement for data sources. Such a buffer-bandwidth tradeoff is shown in Figure 6.4 for $N = 1$, 10 and 20. Notice

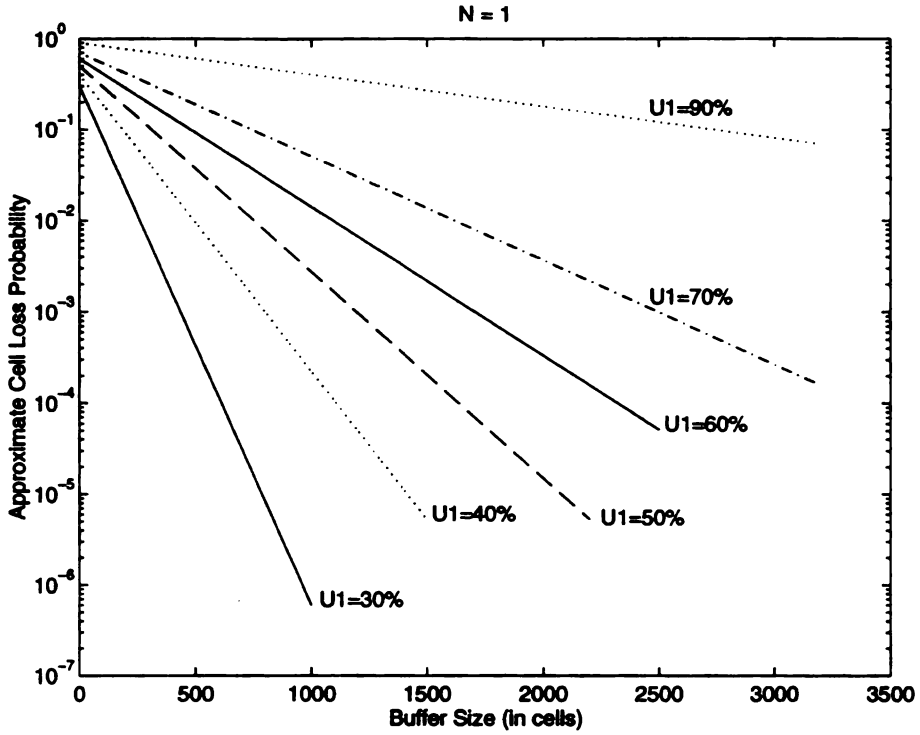

Figure 6.2: Loss probability versus buffer size for data traffic with $N_1 = 1$.

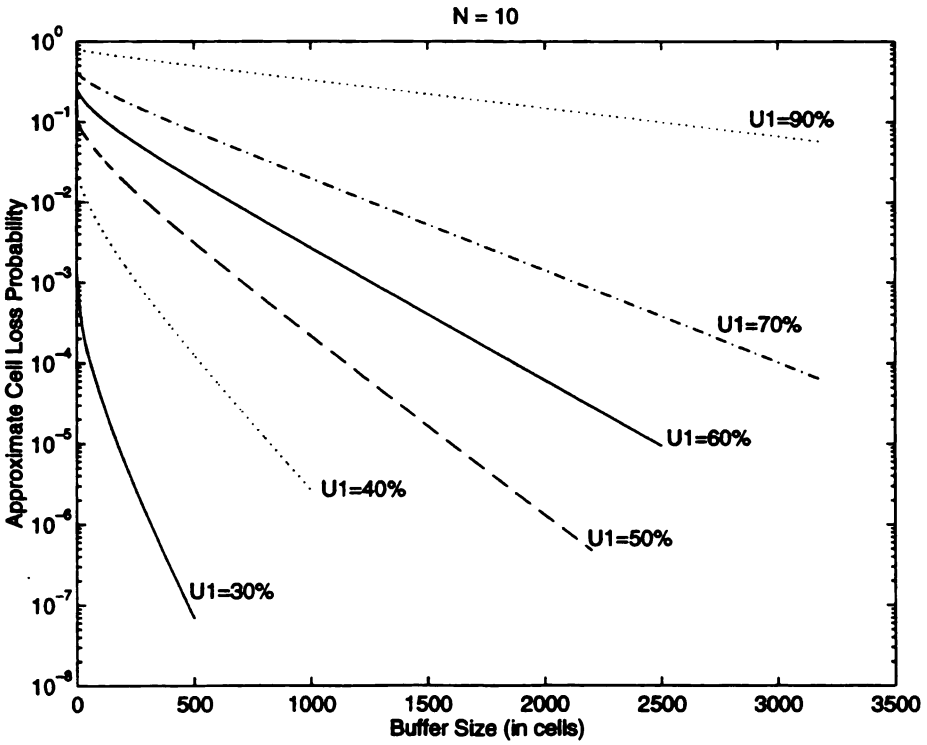

Figure 6.3: Loss probability versus buffer size for data traffic with $N_1 = 10$.

that the Y-axis in this figure indicates the buffer size per source, while the X-axis indicates the offered load, or equivalently, the inverse of bandwidth per source. Any point on each of the three curves will result in a loss rate of 10^{-3}. The choice of a particular buffer-bandwidth pair should be dictated by the available amount of buffer and bandwidth resources as well as their associated costs. Notice that the larger the number of multiplexed streams the smaller are the required amounts of buffer space and bandwidth. A considerable amount of buffers is needed when $N_1 = 1$. Multiplexing significantly reduces the required amount of buffer space per source. This reduction is particularly obvious when going from $N_1 = 1$ to $N_1 = 10$.

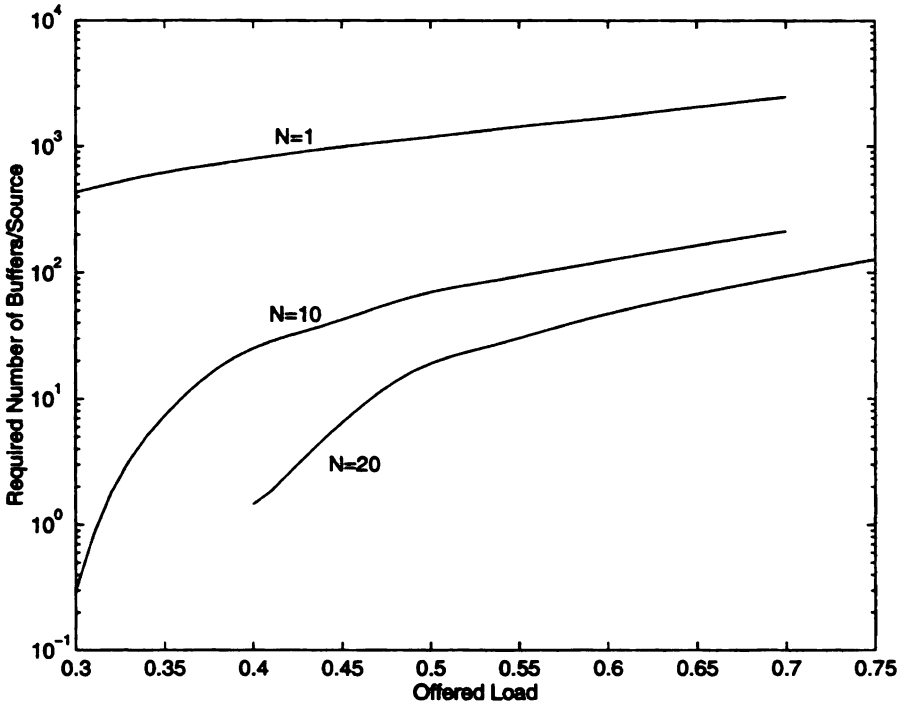

Figure 6.4: Buffer-bandwidth tradeoff for data traffic.

6.3.2 Buffer-Bandwidth Tradeoff for Voice Traffic

Similar to data traffic, the cells loss rate for voice traffic can be computed analytically using the fluid approach. The relationship between the loss rate and the buffer size

can be approximated by the complementary distribution for the content of Buffer 2, as shown in Figures 6.5 and 6.6 for $N_2 = 1$ and $N_2 = 10$, respectively.

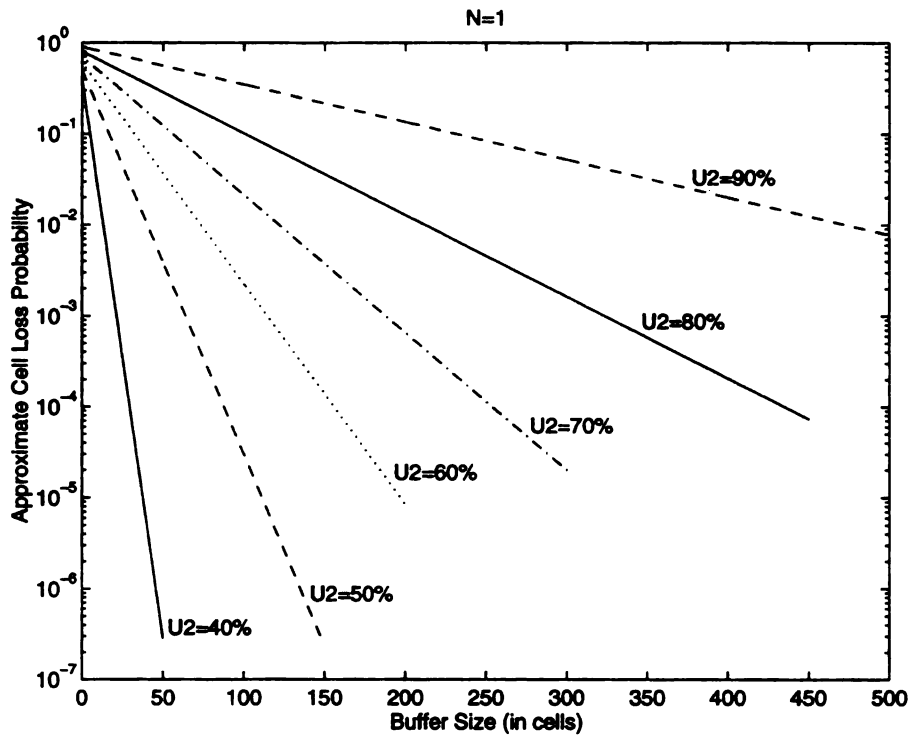

Figure 6.5: Loss probability versus buffer size for voice traffic with $N_2 = 1$.

For voice traffic the maximum offered load that guarantees a zero loss rate without any buffering is given by:

$$U_2^* = \frac{\tau_{ON}^{(2)}}{\tau_{ON}^{(2)} + \tau_{OFF}^{(2)}} = \frac{352 \text{ msec}}{352 \text{ msec} + 650 \text{ msec}} \approx 35\% \tag{6.3}$$

By comparing Figures 6.5 and 6.5 to Figures 6.2 and 6.3, it can be observed that at a given offered load the loss rate for data traffic is much higher than that of voice traffic. This is because of the larger burstiness of a data stream, which can be measured by the ratio of the peak to mean arrival rates (or equivalently, the inverse of U_1^*).

The optimal amount of resources which guarantees ϵ_2 for voice streams is shown

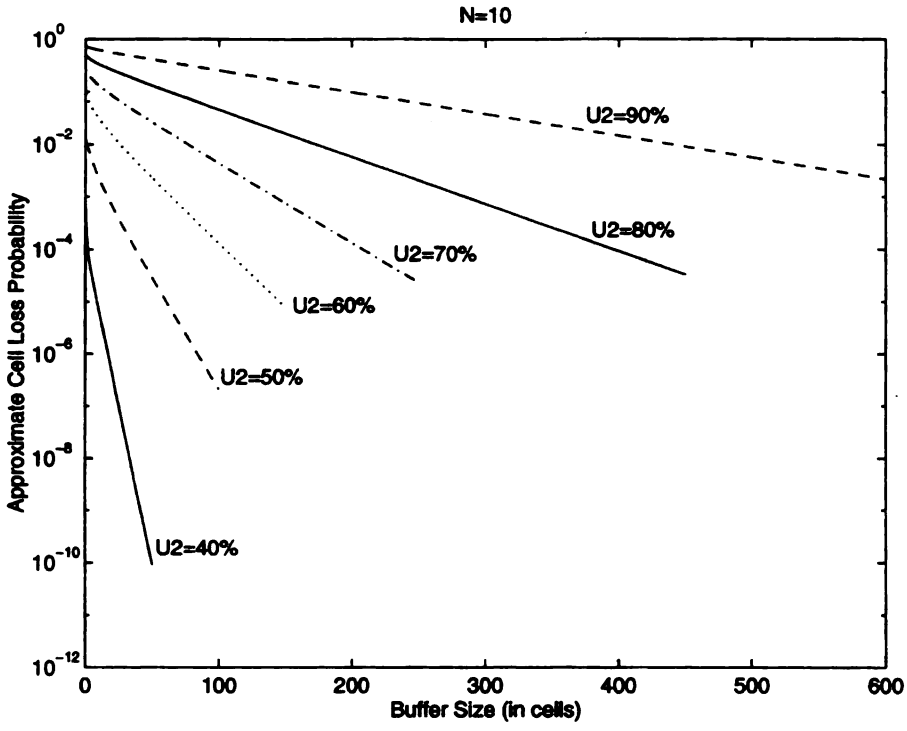

Figure 6.6: Loss probability versus buffer size for voice traffic with $N_2 = 10$.

in Figure 6.7 (the solid curves) for $N_2 = 1$, 10 and 20. Again, any point on each of the three solid curves will guarantee ϵ_2. Clearly, increasing N_2 reduces the required amount of buffers and bandwidth per source. In addition to the loss requirement, voice cells require a maximum delay of $d_2 \leq D_2$. For $U_2 > 35\%$, the maximum delay is determined by U_2 and the buffer size (B_2):

$$d_2 = \begin{cases} \frac{B_2}{C_2}, & \text{if } U_2 > 35\% \\ 0, & \text{if } U_2 \leq 35\% \end{cases} \tag{6.4}$$

But

$$U_2 = \frac{\left(\frac{\tau_{ON}^{(2)}}{\tau_{ON}^{(2)}+\tau_{OFF}^{(2)}}\right)\lambda_2 N_2}{C_2} \tag{6.5}$$

Thus, the buffer-bandwidth tradeoff that guarantees $d_2 \leq D_2$ is:

$$\left(\frac{1}{0.35\lambda_2}\right)\left(\frac{B_2}{N_2}\right) U_2 \leq D_2 \tag{6.6}$$

Using 0.352 sec as the time unit and an ATM cell as the data unit, the buffer-bandwidth tradeoff curves that guarantee the delay requirement are shown by the dotted lines in Figure 6.7, for $D_2 = 0.1$ and $D_2 = 1$ time units (notice that each curve is valid for any N_2). For a given number of voice sources, the buffer-bandwidth operating region should be on the segment of the solid curve associated with N_2, and which lies below the dotted delay curve (to guarantee the delay requirement).

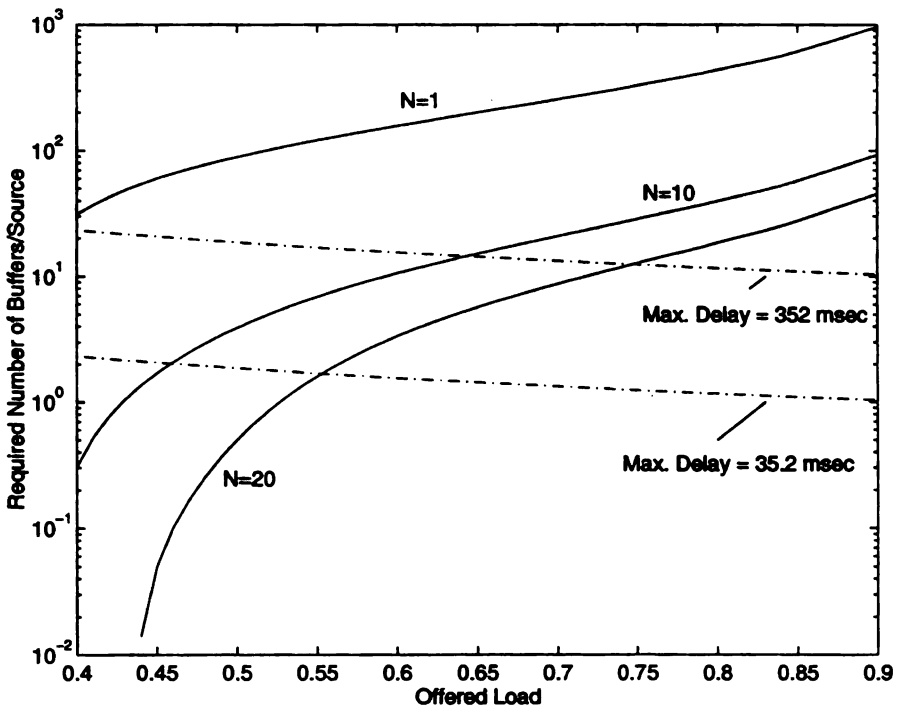

Figure 6.7: Buffer-bandwidth tradeoff for voice traffic.

6.3.3 Buffer-Bandwidth Tradeoff for Video Traffic

In the case of video traffic, obtaining the buffer-bandwidth tradeoff is more difficult than the previous two cases. First, we can only rely on simulations to obtain the relationships between cell loss rates and the amount of resources. Second and more importantly, the dimensioning problem involves, in addition to the buffer size, the threshold (T) which partitions the buffer pool into two portions: one that accommodates all types of cells (i.e., I, P, and B) and the other is exclusive to the high priority cells (i.e., I cells). Our goal in this section is to describe how the buffer size B_3, the threshold value T, and the offered video load U_3 can be optimized to provide the requested QoS requirements ($\epsilon_3^{(1)}$, $\epsilon_3^{(2)}$, and d_3). The results to be described below were obtained using $N_3 = 10$. The same treatment can be carried for any number of video streams.

To obtain the buffer-bandwidth tradeoff curves, consider first the interaction between B_3 and T at a given offered load. In theory, increasing T at a fixed B_3 should result in decreasing $\epsilon_3^{(2)}$ and increasing $\epsilon_3^{(1)}$ (since less fraction of the total buffer space is exclusively reserved for I cells). On the other hand, fixing T and increasing B_3 should clearly decrease $\epsilon_3^{(1)}$ and, less obviously, increase $\epsilon_3^{(2)}$. This is true because an increase in B_3 means higher probability of admitting I cells to the buffer, which in turn will adversely affect the loss rate for low priority cells. However, if $\epsilon_3^{(1)}$ and $\epsilon_3^{(2)}$ are orders of magnitude different, then the impact of increasing B_3 (at fixed T) on $\epsilon_3^{(2)}$ is negligible. We verified our intuition by conducting several experiments in which B_3 was varied at some fixed T. The output of these experiments is shown in Figure 6.8 using $U = 40\%$ and $U = 80\%$, and different values of T.

The above observation allows us to decompose the dimensioning problem into three stages. In the first stage, T is optimized to satisfy the $\epsilon_3^{(2)} \leq 10^{-2}$ requirement using an arbitrary large buffer size. Given the optimum T from the first stage, say T^*, the buffer size (B_3) is optimized in the second stage to satisfy the $\epsilon_3^{(1)} \leq 10^{-4}$

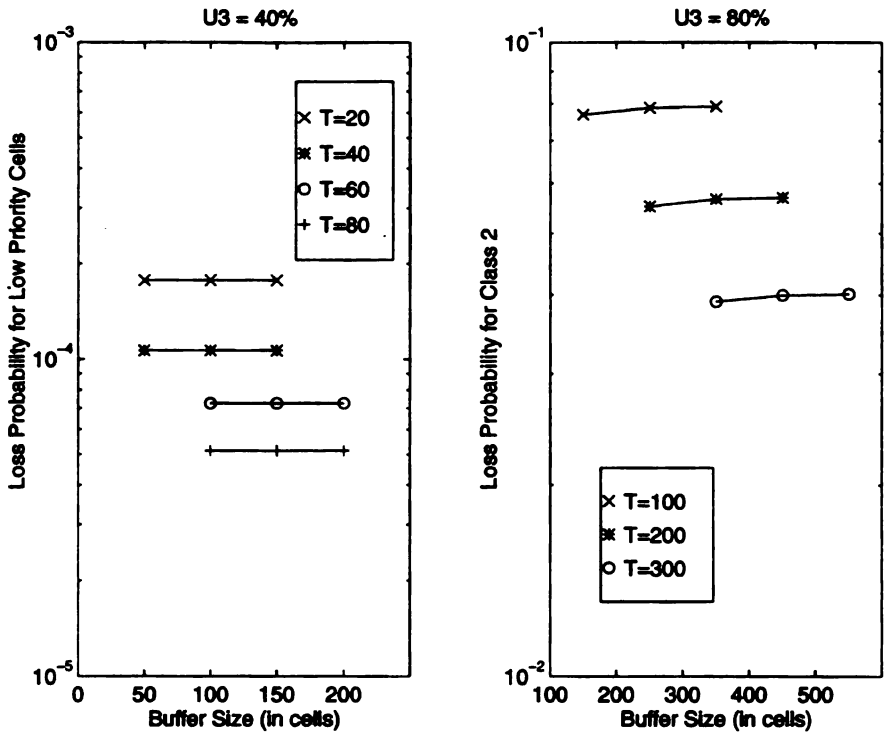

Figure 6.8: Loss probability for low priority (P and B) cells versus buffer size.

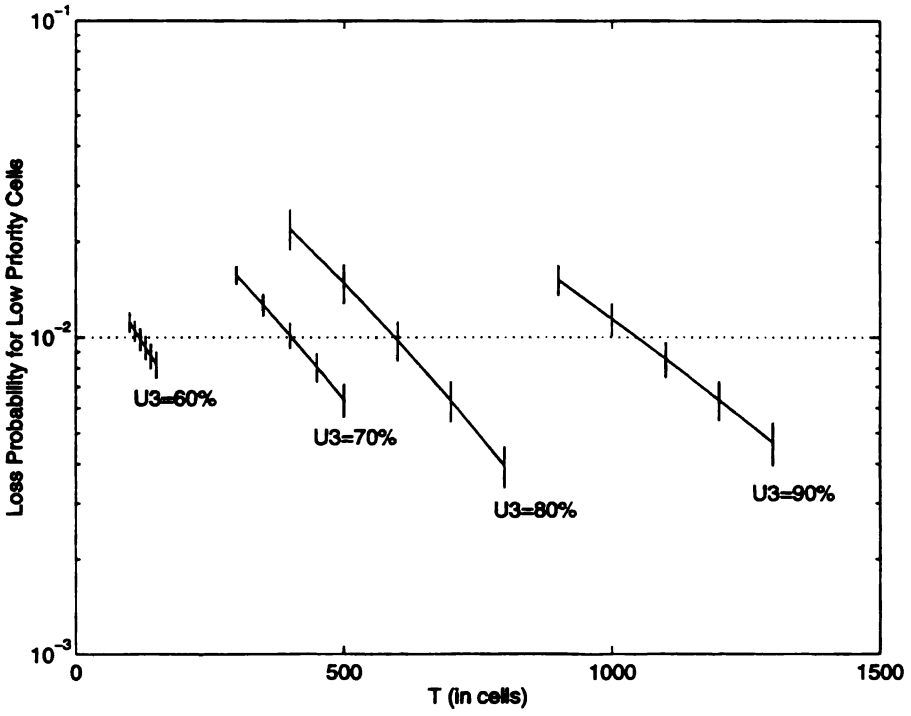

Figure 6.9: Loss probability for low priority (P and B) cells versus threshold.

requirement. Finally, an additional buffer-bandwidth relationship is drawn to satisfy the delay requirement $d_3 \leq D_3$.

Figure 6.9 depicts the cell loss rate for low priority cells (P and B) versus T using arbitrary large values for B_3 and at different offered loads. Each experiment was replicated 10 times, each time using a different seed for the random number generator. The 95% confidence intervals are shown by the short vertical lines in the graph. The values for B_3 which were used are: 200, 500, 600, 1000, and 1500 cells, at the corresponding offered loads: $U_3 = 0.6,\ 0.7,\ 0.8$, and 0.9. Clearly, increasing U_3 requires a drastic increase in T to maintain a cell loss rate in the range of 10^{-2}.

For video traffic, the maximum offered load which guarantees a zero loss rate at $B_3 = 0$ is given by:

$$U_3^* = \frac{\text{average arrival rate of a stream}}{\text{peak arrival rate of a stream}} = \frac{109.5 \text{ cells/frame}}{894 \text{ cells/frame}} = 12.2\% \qquad (6.7)$$

Once T^* is computed, we can now dimension the size of the buffer to satisfy $\epsilon_3^{(1)}$. Using $T = T^*$ at different U_3, simulations were conducted in which B_3 was varied. Figure 6.10 depicts the loss rate for high priority cells ($\epsilon_3^{(1)}$) versus B_3 when $T = T^*$. In all of these simulations, $\epsilon_3^{(2)}$ was constantly observed to be around 10^{-2}, as expected. From Figure 6.10 the minimum value for B_3, say B_3^*, that satisfies $\epsilon_3^{(1)} \leq 10^{-4}$ can be computed. These values are given in Table 6.1.

U_3	T_3^*	B_3^*
40%	1	3
50%	3	7
60%	118	195
70%	404	487
80%	625	760
90%	1100	1230

Table 6.1: Optimal values for T_3 and B_3 that guarantee the QoS requirements.

To satisfy the delay requirement $d_3 \leq D_3$, a similar buffer-bandwidth relationship

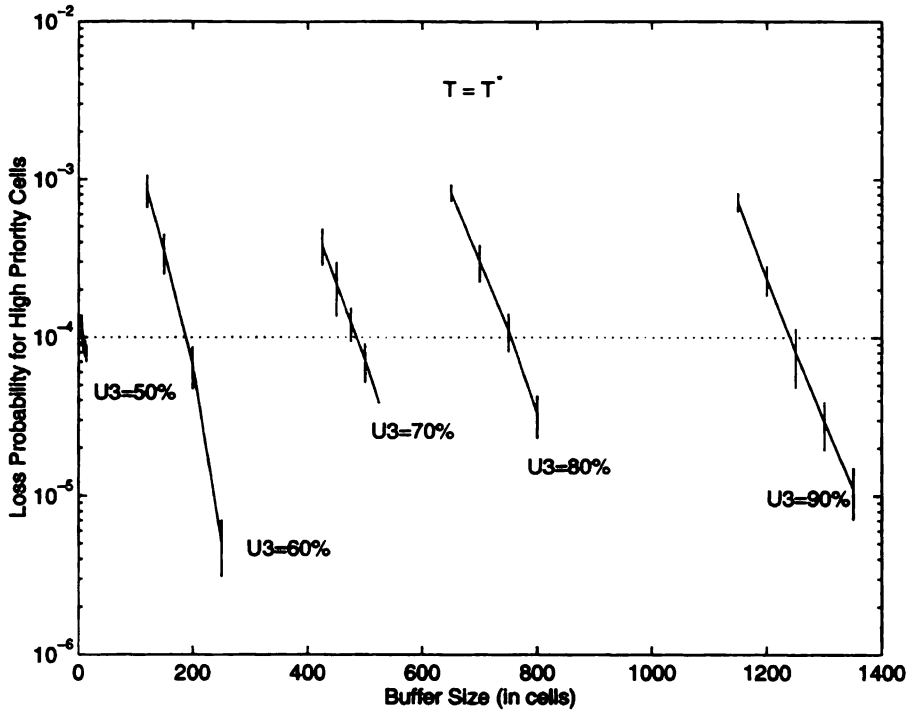

Figure 6.10: Loss probability for high priority (I) cells versus buffer size at $T = T^*$.

to the data and voice cases can be obtained as follows:

$$\left(\frac{1}{109.5}\right)\left(\frac{B_3}{N_3}\right) U_3 \leq D_3, \quad \text{for } U_3 > U_3^* \tag{6.8}$$

Using (6.8) and the results in Table 6.1, the buffer-bandwidth tradeoff curves for video traffic can be obtained as shown in Figure 6.11.

6.3.4 Performance under a Work-Conserving Discipline

In the previous sections, buffer and bandwidth resources in the NTCD/MB mechanism were optimized to provide the given set of QoS requirements. However, to maintain tractable results, the optimization was carried under the assumption of non-work-conserving service discipline. From a practical point of view, it is often desirable to use a work-conserving discipline when cells are output from multiple buffers.

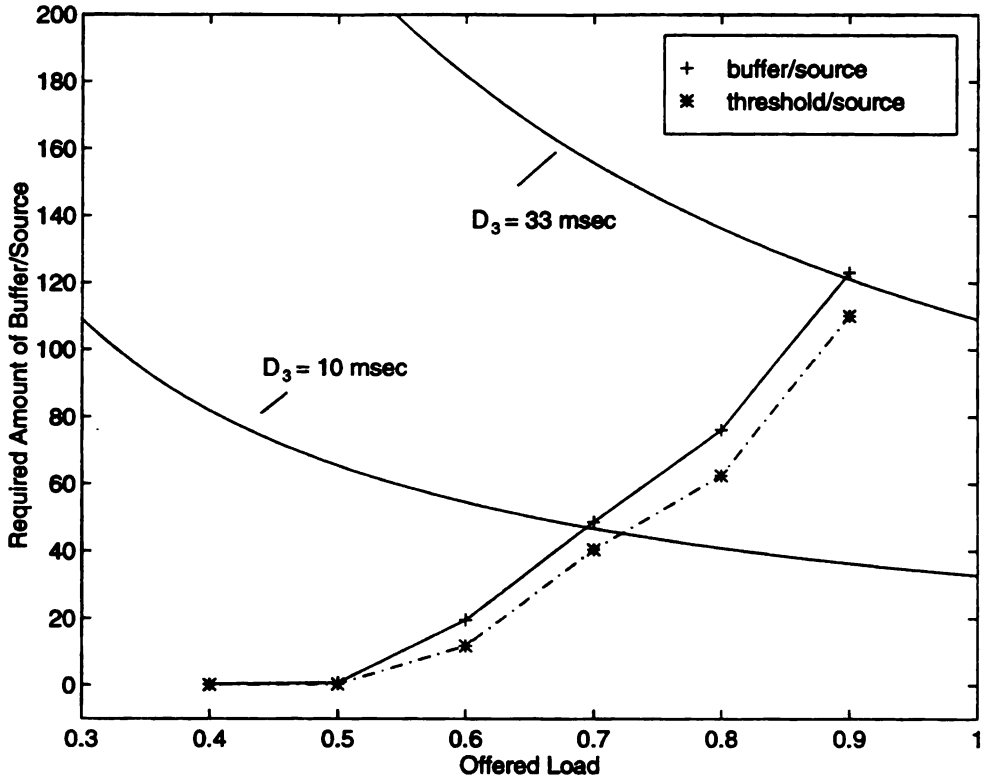

Figure 6.11: Buffer-bandwidth tradeoff for video traffic.

Intuitively, a work-conserving discipline should enhance the overall performance, or equivalently, maintain the same performance in a non-work-conserving discipline but using less amount of resources. Unfortunately, resource optimization in the underlying 3-buffer system is not analytically tractable in the case of a work-conserving discipline. To provide efficient resource allocation for such a buffer system, we start with the optimal values for buffers and bandwidth in the non-work-conserving version. These values will serve as upper bounds to the optimal amount of resources in the work-conserving case. We then evaluate the performance using these values. If the performance is considerably better than the required QoS requirements, the values for buffer sizes and allocated bandwidth can be reduced until the performance is closer to the requested QoS requirements. In the following, we illustrate the above procedure by an example.

Let $N_1 = N_2 = N_3 = 10$ sources. Assume the same QoS requirements as before

with $D_2 = 352$ msec and $D_3 = 33$ msec. Suppose that the target offered loads for each type of traffic are: $U_1 = 50\%$, $U_2 = 60\%$, and $U_3 = 70\%$. From these values, we can compute the service rates C_1, C_2, and C_3 using, for example, (6.1). Accordingly, $C_1 = 943$, $C_2 = 443$, and $C_3 = 46928$ (all in cells/sec). The total link capacity is $C = C_1 + C_2 + C_3 = 48314$ cells/sec. Clearly, the amount of bandwidth to be allocated for video traffic is much larger than the bandwidth allocated for data or voice traffic. The large difference in the allocated bandwidth makes the gains from a work-conserving discipline vary from one traffic to another. For data and voice streams, a work-conserving discipline will drastically improve the queueing performance (compared to a non-work-conserving discipline). The reason is that when no video cells are in Buffer 3, a huge residual bandwidth (C_3) will be used to switch voice and data cells in the other two buffers. Video traffic, on the other hand, will experience only a slightly better performance when the relatively small amounts of bandwidth (C_1 and C_2) are added to the service rate of Buffer 3.

Typically, the NTCD/MB mechanism is implemented using a weighted round-robin discipline. This requires that the total service rate be transformed into an integer number of data units (q) per service cycle, and that the service rates guaranteed to individual buffers be also some integers of data units, say q_1, q_2, and q_3 (with $q = q_1 + q_2 + q_3$). In theory, these integer quantities can be the same as the above values for C, C_1, C_2, and C_3. However, this results in large blocking times at a given buffer when the server is busy switching cells from another buffer. Moreover, large counters would be needed to keep track of the cell counts. To reduce the blocking times, it is better to slightly adjust the service rates so that the q_is are kept as small as possible. We adjust the service rates to become: $\tilde{C}_1 = 1000$, $\tilde{C}_2 = 500$, and $\tilde{C}_3 = 47000$ cells/sec. Based on these rates, a service cycle may consist of serving a

maximum of 100 cells (i.e., $q = 100$) from the three buffers such that:

$$\begin{aligned} q_1 &= 2 \text{ cells} \\ q_2 &= 1 \text{ cells} \\ q_3 &= 97 \text{ cells} \end{aligned}$$

The adjusted service rates correspond to $U_1 = 0.47$, $U_2 = 0.53$, and $U_3 = 0.68$. At these loads, the optimal buffer sizes (and the optimal T for the third buffer) under a non-conserving discipline can be obtained from the tradeoff curves in Figures 6.4, 6.7, and 6.11, and were found to be:

$$\begin{aligned} \hat{B}_1 &\approx 500 \text{ cells} \\ \hat{B}_2 &\approx 50 \text{ cells} \\ \hat{B}_3 &\approx 480 \text{ cells} \\ \hat{T} &\approx 380 \text{ cells} \end{aligned}$$

Note, however, that these values are not necessarily optimal under a work-conserving discipline. Simulations of the (work-conserving) NTCD/MB mechanism were conducted used the above values for the buffers and the threshold. The above values for q, q_1, q_2, and q_3 were assumed. From 20 replications, the cell loss rates for the various types of cells were found to be as follows (confidence intervals are indicated as well):

$$\epsilon_1 = 0 \pm 0 \tag{6.9}$$

$$\epsilon_2 = 0 \pm 0 \tag{6.10}$$

$$\epsilon_3^{(1)} = 8.36 \times 10^{-5} \pm 3.4 \times 10^{-5} \tag{6.11}$$

$$\epsilon_3^{(2)} = 1.09 \times 10^{-2} \pm 3.2 \times 10^{-3} \tag{6.12}$$

It is clear that the requested QoS are being satisfied (the delay requirements are being met by the choice of the buffer sizes). Moreover, the cell loss rates for data and voice streams are much smaller than needed, whereas the loss rates for video streams are, as expected, within the range of the requested QoS. Since it is sufficient to only provide the requested QoS, we may further reduce the sizes of the data and voice buffers until their associated cell loss rates are in the ranges of the requested QoS. Simulations were used in which the values for B_1 and B_2 were varied with B_3 and T kept constant at their previous values. The results are summarized in Figure 6.12.

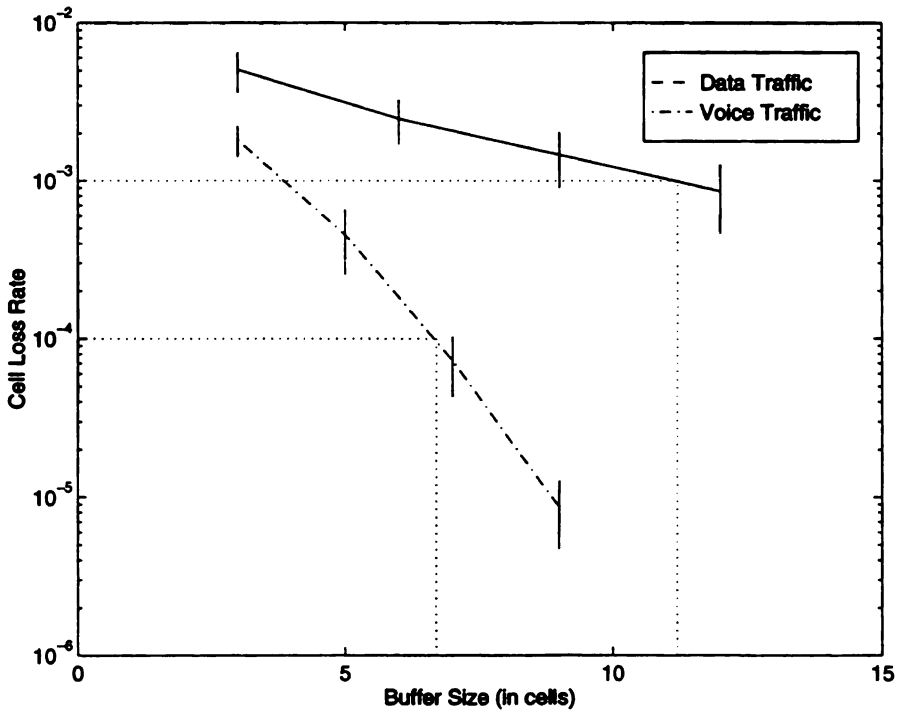

Figure 6.12: Cell loss rates versus buffer sizes for data and voice traffic.

From this figure, we found the minimum values for B_1 and B_2 that guarantee the requested QoS requirements to be:

$$B_1^* \approx 12 \text{ cells}$$

$$B_2^* \approx 8 \text{ cells}$$

which are considerably less than the values obtained based on a non-conserving service discipline.

Chapter 7

Summary and Future Research

7.1 Summary

In this thesis we introduced a new buffer management scheme for multimedia traffic in ATM networks. This scheme, referred to as NTCD/MB, is designed to support multiple grades of service that are often required in a heterogeneous multimedia traffic environment. NTCD/MB combines the use of loss and delay priority of the incoming traffic to support several sets of QoS requirements. It exhibits a simple cell admission and scheduling disciplines, which makes the mechanism easy to implement in hardware. Moreover, the mechanism provides *guaranteed* performance to several types of traffic while efficiently utilizing the available resources. We propose the use of NTCD/MB for buffer management at ATM switching and multiplexing nodes.

We outlined the general structure of the NTCD/MB mechanism and evaluated its performance under various traffic scenarios. Preliminary investigation of the performance of NTCD/MB was carried via simulations assuming data and voice traffic streams. The MMPP model was used to characterize each type of traffic. Different cases were considered to demonstrate the flexibility of the NTCD/MB mechanism to adapt to various traffic mixes and different sets of performance requirements.

A complete analysis of the queueing performance of the NTCD/MB was provided using the fluid representation for the input streams. This analytical study was conducted for a particular structure of the NTCD/MB mechanism that consists of two buffers and a threshold. However, the same procedure can be applied to the analysis of any structure of the NTCD/MB mechanism. From this analysis, we obtained expressions for the steady-state probability distribution for the content of each buffer, the cell loss rate, the throughput, and an approximate waiting time distribution for each traffic type. Our approach can be used to analyze any dependent system of buffers in which the input traffic can be conveniently characterized by separable Markov-modulated fluid processes. Numerical results based on the analysis were supported by similar ones obtained via simulations.

The fluid approximation is known to be particularly appropriate for data and voice streams. It can also be used to characterize compressed video streams that are generated by certain types of video encoders. Recent advances in video coding resulted in compression encoders that generate very complicated traffic patterns. Such patterns can not be adequately approximated by fluid models. Motivated by the desire to evaluate the performance of NTCD/MB under MPEG-compressed video streams (possibly along with other data and voice streams), we studied the characteristics of MPEG-compressed movies. Two models were developed to characterize an MPEG stream. The first model is based on the compression pattern and the distributions of the bit rates from the three types of MPEG frames. Bit rate variations due to scene changes are additionally incorporated in the second model. The two models capture several statistical properties of the empirical video data. More importantly, the queueing performance under synthetic streams generated by either model was shown to be very close to the trace-driven performance. It was observed that the queueing performance under the second model is always slightly more conservative compared to the trace-driven performance. Since the main idea behind NTCD/MB

is to guarantee the performance, the second model was chosen to characterize video streams in the subsequent studies.

Through a proof-of-concept study, we described (in *Chapter 6*) how to efficiently allocate buffer and bandwidth resources in the NTCD/MB mechanism. In that study, we combined our previous analytical and simulation results to compute the optimal buffer-bandwidth pairs that guarantee a set of requested QoS requirements under a non-work-conserving service discipline. Buffer-bandwidth tradeoff curves were established for the three examined types of traffic: data, voice, and video (MPEG-compressed). These curves were then used to provide conservative estimates for the optimal amounts of resources under a work-conserving discipline, which were subsequently computed via simulations.

7.2 Future Research

7.2.1 Generalizing the Proposed Video Models

The traffic models developed in *Chapter 5* were based on one trace of empirical data and also using a particular compression pattern. This is due to the excessive amount of time that is required to capture and compress video frames (it took six weeks to digitize and store the frames and three weeks to compress them). Parameters of both models were obtained by matching certain statistics of the model to their correspondents in the empirical trace. Hence, these parameters are specific to the examined video trace, i.e., a different movie is expected to result in different values for these parameters. Moreover, using a different compression pattern to generate an MPEG steam may also result in different values for the model parameters.

Our objective is to develop a procedure to estimate the parameters of each model to fit a given movie. Only few information about the movie should be needed to estimate the values for the parameters of either model. Such information could include

the compression pattern, the dimensions of the frame, and the "level of dynamics" of the movie. This last requirement implies that movies should be classified into a small number of classes based on their level of scene activity. Movies belonging to a given class and that have the same compression pattern and image dimensions will probably result in similar values for model parameters. Clearly, to achieve our goal, we need to consider a large number of movies to be individually compressed according to various compression patterns.

7.2.2 Analyzing the Queueing Performance of MPEG Streams

The queueing performance for MPEG streams was obtained in this thesis via simulations. For cell loss rates below 10^{-6}, simulations become computationally impractical. Analytical results, if tractable, provide a very efficient and inexpensive means of obtaining very small loss rate performance. Unfortunately, exact queueing analysis does not seem possible for any of the proposed models. Instead, approximate bounding approaches are more promising for analysis purposes. We intend to further explore this possibility. One may, for example, assume that MPEG frames of a certain type (I, P, or B) have the size of the largest frame of their type. The three maximum sizes may be provided *apriori* or estimated. Accordingly, a video stream is approximately described by the three maxima and the compression pattern. This characterization has more potential for queueing analysis than a more accurate characterization that is provided by any of the proposed models.

Bibliography

[1] T. G. Alliance. The U.S. HDTV standard: the Grand Alliance (special report on digital TV). *IEEE Spectrum*, pages 36–45, Apr. 1995.

[2] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic theory of a data-handling system with multiple sources. *Bell System Technical Journal*, 61(8):1871–1894, Oct. 1982.

[3] J. J. Bae and T. Suda. Survey of traffic control schemes and protocols in ATM networks. *Proceedings of the IEEE*, 79(2):1871–1894, Feb. 1991.

[4] J. Banks and J. S. Carson. *Discrete-Event System Simulation.* Prentice-Hall, Inc., first edition, 1984.

[5] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forcasting and Control.* Holden-Day, Inc., 1976. (Revised Edition).

[6] A. W. Bragg and W. Chou. Analytic models and characteristics of video traffic in high speed networks. In *Proceedings of International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '94)*, pages 68–73, Durhan, North Carolina, Jan. 31 – Feb. 2 1994.

[7] S. K. Chan and A. L. Garcia. Analysis of cell inter-arrival from VBR video codecs. In *Proc. of IEEE INFOCOM '94*, volume 1, pages 350–357, 1994.

[8] T. M. Chen, J. Walrand, and D. G. Messerschmitt. Dynamic priority protocols for packet voice. *IEEE Journal on Selected Areas in Communications*, 7(7), June 1989.

[9] R. Chipalkatti, J. F. Kurose, and D. Towsley. Scheduling policies for real-time and nonreal-time traffic in a statistical multiplexer. In *Proceedings of IEEE INFOCOM '89*, pages 774–783, 1989.

[10] M. De Prycker. Evolution from ISDN to BISDN: a logical step towards ATM. *Computer Communications*, 12(3):141–146, 1989.

[11] S. Dravida and K. Sriram. End-to-end performance models for variable bit rate voice over tandem links in packet networks. In *Proc. of IEEE INFOCOM '89*, pages 1089–1095, 1989.

[12] A. I. Elwalid and D. Mitra. Fluid models for the analysis and design of statistical multiplexing with loss priorities on multiple classes of bursty traffic. In *Proceedings of IEEE INFOCOM '92*, pages 3c.4.1–3c.4.11, 1992.

[13] A. I. Elwalid, D. Mitra, and T. E. Stern. Statistical multiplexing of Markov modulated sources: Theory and computational algorithms. In *Proceedings of 13th International Teletraffic Congress (ITC)*, June 1992.

[14] W. Fischer and K. Meier-Hellstern. The markov-modulated poisson process (MMPP) cookbook. *Performance Evaluation*, 1(18):149–171, 1993.

[15] V. S. Frost and B. Melamed. Traffic modeling for telecommunications networks. *IEEE Communications Magazine*, 32(3):70–81, Mar. 1994.

[16] J. Garcia and O. Casals. Priorities in ATM networks. In *Proc. of NATO Workshop on Architecture and Performance issues of high capacity local and metropolitan networks*, June 1990.

[17] M. W. Garrett and W. Willinger. Analysis, modeling, and generation of self-similar VBR video traffic. *Computer Communication Review (SIGCOMM'94 Conference Proceedings)*, pages 269–280, Sept. 1994.

[18] M. Ghanbari. Two-layer coding of video signals for VBR networks. *IEEE Journal on Selected Areas in Communications*, 7(5):771–781, June 1989.

[19] D. J. Goodman. Embedded DPCM for variable bit rate transmission. *IEEE Transaction on Communications*, 28:1040–1046, July 1980.

[20] A. Gravey and G. Hebuterne. Analysis of a priority queue with delay and/or loss sensitive customers. In *Proc. of ITC Broadband Seminar*, Oct. 1990.

[21] R. Grunenfelder, J. P. Cosmos, S. Manthorpe, and A. Odinma-Okafor. Characterization of video codecs as autoregressive moving average processes and related queueing system performance. *IEEE Journal on Selected Areas in Communications*, 9(3):284–293, Apr. 1991.

[22] H. Heffes and D. M. Lucantoni. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE Journal on Selected Areas in Communications*, SAC-4(6):856–868, Sept. 1986.

[23] D. P. Heyman and T. V. Lakshman. Source models for VBR broadcast-video traffic. In *Proc. of IEEE INFOCOM '94*, volume 2, pages 664–671, 1994.

[24] D. P. Heyman, A. Tabatabai, and T. V. Lakshman. Statistical analysis and simulation study of video teleconferencing traffic in ATM networks. *IEEE Trans. on Circuits and Systems for Video Technology*, 2(1):49–59, Mar. 1992.

[25] T. Hou and A. Wong. Queueing analysis for ATM switching of mixed continuous-bit-rate and bursty traffic. In *Proc. of IEEE INFOCOM '90*, pages 660–667, 1990.

[26] ISO/MPEG. ISO CD11172-2: Coding of moving pictures and associated audio for digital storage media at up to about 1.5 mbits/s, Nov. 1991.

[27] ISO/MPEG II. ISO CD11172-2: Coding of moving pictures and associated audio, Dec. 1992.

[28] D. L. Jagerman and B. Melamed. The transition and autocorrelation structure of TES processes part I: General theory. *Stochastic Models*, 8(2):193–219, 1992.

[29] D. L. Jagerman and B. Melamed. The transition and autocorrelation structure of TES processes part II: Special cases. *Stochastic Models*, 8(3):499–527, 1992.

[30] R. Jain and S. Routhier. Packet train – measurements and a new model for computer network traffic. *IEEE Journal on Selected Areas in Communications*, Sept. 1986.

[31] N. S. Jayant and S. W. Christensen. Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation procedure. *IEEE Transaction on Communications*, 29:101–109, Feb. 1981.

[32] F. Kishino, K. Manabe, Y. Hayashih, and H. Yasuda. Variable bit-rate coding of video signals for ATM networks. *IEEE Journal on Selected Areas in Communications*, 7(5):801–806, June 1989.

[33] L. Kosten. Stochastic theory of data-handling systems with groups of multiple sources. *Performance of Computer-Communication Systems*, pages 321–331, 1984.

[34] H. Kroner. Comparative performance study of space priority mechanisms for ATM channels. In *Proc. of IEEE INFOCOM '90*, pages 1136–1143, 1990.

[35] H. Kroner, G. Hebuterne, P. Boyer, and A. Gravey. Priority management in ATM switching nodes. *IEEE Journal on Selected Areas in Communications*, 9(3):418–427, Apr. 1991.

[36] M. Krunz and H. Hughes. Analysis of a Markov-modulated fluid flow model for multimedia traffic with loss and delay priorities. *Journal of High Speed Networks*, 3(3):309–329, 1994. IOS Press.

[37] M. Krunz and H. Hughes. A performance study of loss priorities for MPEG video traffic. In *in Proc. of IEEE ICC '95 Conference*, June 1995.

[38] M. Krunz and H. Hughes. A traffic model for MPEG-coded VBR streams. In *Proc. of the ACM SIGMETRICS/PERFORMANCE '95 Conference*, pages 47–55, May 1995.

[39] M. Krunz, H. Hughes, and P. Yegani. Congestion control in ATM networks using multiple buffers and priority mechanisms. In *Proc. of the Summer Computer Simulations Conference*, pages 566–571, Boston, MA, July 1993.

[40] M. Krunz, H. Hughes, and P. Yegani. Traffic control in ATM switching using multiple buffers and nested threshold cell discarding. In *Proc. of 2'nd Int. Conf. on Computer Communications and Networks (IC3N)*, pages 121–127, San Diego, California, June 1993.

[41] M. Krunz, H. Hughes, and P. Yegani. Design and analysis of a buffer management scheme for multimedia traffic with loss and delay priorities. In *Proc. of the IEEE GLOBECOM '94*, pages 1560–1564, San Francisco, CA, Nov. 1994.

[42] M. Krunz, R. Sass, and H. Hughes. Statistical characteristics and multiplexing of MPEG streams. In *Proc. of the IEEE INFOCOM '95 Conference*, pages 455–462, Boston, Apr. 1995.

[43] M. Krunz, R. Sass, and H. Hughes. A study of VBR MPEG-coded video traffic and associated multiplexing performance. *Computer Systems Science and Engineering Journal*, 1995. CRL Publishing Ltd. (in press).

[44] J. Kurose. Open issues and challenges in providing quality of service guarantees in high-speed networks. *Computer Communication Review, ACM/SIGCOMM*, 23(1):6–15, Jan. 1993.

[45] A. M. Law and W. D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, Inc., second edition, 1991.

[46] A. A. Lazar, G. Pacifici, and D. E. Pendarakis. Modeling video sources for real-time scheduling. Technical Report 324–93–03, Columbia University, Department of Electrical Engineering and Center for Telecommunications Research, Apr. 1993.

[47] D. Le Gall. MPEG: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):47–58, Apr. 1991.

[48] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1):1–15, Feb. 1994.

[49] Y. Lim and J. Kobza. Analysis of a delay-dependent priority discipline in a multiclass traffic packet switching node. In *Proc. of IEEE INFOCOM '88*, pages 9A.4.1–9A.4.1.10, 1988.

[50] A. Y.-M. Lin and J. A. Silvester. Priority queueing strategies and buffer allocation protocols for traffic control at an ATM integrated broadband switching system. *IEEE Journal on Selected Areas in Communications*, 9(9):1524–1536, Dec. 1991.

[51] M. L. Luhanga. A fluid approximation model of an integrated packet voice and data multiplexer. In *Proc. of IEEE INFOCOM '88*, pages 7A.4.1–7A.4.6, 1988.

[52] B. Maglaris et al. Performance models of statistical multiplexing in packet video communications. *IEEE Journal on Selected Areas in Communications*, 36(7):834–844, July 1988.

[53] N. M. Marafih, Y.-Q. Zhang, and R. L. Pickholtz. Modeling and queueing analysis of variable-bit-rate coded video sources in ATM networks. *IEEE Trans. on Circuits and Systems for Video Technology*, 4(2):121–128, Apr. 1994.

[54] B. Melamed. TES: A class of methods for generating autocorrelated uniform variates. *ORSA Journal on Computing*, 3(4):317–329, 1991.

[55] S. E. Minzer. Broadband ISDN and Asynchronous Transfer Mode (ATM). *IEEE Communications Magazine*, 27(9):17–24, Sept. 1989.

[56] D. Mitra. Stochastic theory of a fluid model of producers and consumers coupled by a buffer. *Advances in Applied Probability*, 20:646–676, 1988.

[57] R. W. Muise, T. J. Scheonfeld, and G. H. Zimmerman. Experiments with wideband packet technology. In *Proc. of International Zurich Seminar on Digital Communications*, Mar. 1986. (paper D4).

[58] P. Pancha and M. El Zarki. MPEG coding for variable bit rate video transmission. *IEEE Communications Magazine*, 32(5):54–66, May 1994.

[59] D. W. Petr, L. A. DaSliva, Jr., and V. S. Frost. Priority discarding of speech in integrated packet networks. *IEEE Journal on Selected Areas in Communications*, 7(5):644–659, June 1989.

[60] D. W. Petr and V. S. Frost. Nested threshold cell discard for ATM overload control: optimization under cell loss constraints. In *Proc. of IEEE INFOCOM '91*, pages 12A.4.1–12A.4.1.10, Bal Harbour, 1991.

[61] G. Ramamurthy and B. Sengupta. Modeling and analysis of a variable bit rate video multiplexer. In *Proc. of IEEE INFOCOM '92*, volume 2, pages 817–827, 1992.

[62] P. Sen, B. Maglaris, N.-E. Rikli, and D. Anastassiou. Models for packet switching of variable-bit-rate video sources. *IEEE Journal on Selected Areas in Communications*, 7(5):865–869, June 1989.

[63] P. Skelly, M. Schwartz, and S. Dixit. A histogram-based model for video traffic behavior in an ATM multiplexer. *IEEE/ACM Transactions on Networking*, 1(4):446–459, Aug. 1993.

[64] K. Sohraby. On the theory of general ON-OFF sources with applications in high-speed networks. In *Proceedings of IEEE INFOCOM '93*, pages 401–410, 1993.

[65] K. Sriram and W. Whitt. Characterizing superposition arrival processes in packet multiplexers for voice and data. *IEEE Journal on Selected Areas in Communications*, SAC-4(6):833–846, Sept. 1986.

[66] T. E. Stern and A. I. Elwalid. Analysis of separable Markov-modulated rate models for information-handling systems. *Advances in Applied Probability*, 23:105–139, 1991.

[67] R. C. F. Tucker. Accurate method for analysis of a packet-speech multiplexer with limited delay. *IEEE Transaction on Communications*, 36(4):479–483, Apr. 1988.

[68] J. S. Turner and L. F. Wyatt. A packet network architecture for integrated services. In *Proceedings of IEEE GLOBECOM '83*, pages 2.1.1–6, 1983.

[69] D. Verma, H. Zhang, and D. Ferrari. Guaranteeing delay jitter bounds in packet-switched networks. In *Proc. of TriComm '92*, Apr. 1991. (Chapel Hill, NC).

[70] G. Woodruff and R. Kositpaiboon. Multimedia traffic management principles for guaranteed ATM network performance. *IEEE Journal on Selected Areas in Communications*, 8(3):437–446, Apr. 1990.

[71] V. Yau and K. Pawlikowski. ATM buffer overflow control: A Nested Threshold Cell Discarding with Suspended Execution. In *Proc. of Australian Broadband Switching and Services Symp.*, Melbourne, July 1992.

[72] P. Yegani. Performance models for ATM switching of mixed continuous bit rate and bursty traffic with threshold based discarding. In *Proc. of IEEE ICC '92*, pages 1621–1627, June 1992.

[73] P. Yegani, M. Krunz, and H. Hughes. Congestion control schemes in prioritized ATM networks. In *Proc. of IEEE ICC '94*, pages 1169–1173, May 1994.

[74] F. Yegengolu, B. Jabbari, and Y.-Q. Zhang. Motion-classified autoregressive modeling of variable bit rate video. *IEEE Trans. on Circuits and Systems for Video Technology*, 3(2):42–53, Feb. 1993.

[75] N. Yin, S.-Q. Li, and T. E. Stern. Congestion control for packet voice by selective packet discarding. In *Proc. of IEEE GLOBECOM '87*, pages 45.3.1–45.3.4, 1987.

[76] N. Yin, S.-Q. Li, and T. E. Stern. Data performance in an integrated packet voice/data system using voice congestion control. In *Proc. of IEEE GLOBECOM '88*, pages 16.4.1–16.4.5, 1988.

[77] N. Yin, T. E. Stern, and S.-Q. Li. Performance analysis of a priority-oriented packet voice system. In *Proc. of IEEE INFOCOM '87*, pages 856–863, 1987.

[78] J. Zhang. Performance study of Markov modulated fluid flow models with priority traffic. In *Proc. of IEEE INFOCOM '93*, pages 1a.2.1–1a.2.8, 1993.