Performance Analysis of Error Recovery and Congestion Control in High-Speed Networks*

Jaime Jungok Bae
Technical Report No. 91-76

Department of Information and Computer Science
University of California, Irvine
Irvine, CA 92717

* This Technical Report is based on the author's dissertation for the Ph.D. degree.

Abstract of the Dissertation

Performance Analysis of Error Recovery
and Congestion Control in High-Speed Networks

by

Jaime Jungok Bae

Doctor of Philosophy in Information and Computer Science

University of California, Irvine, 1991

Professor Tatsuya Suda, Chair

In the past few years, Broadband Integrated Services Digital Network (B-ISDN) has received increasing attention as a communication architecture capable of supporting multimedia applications. Among the techniques proposed to implement B-ISDN, Asynchronous Transfer Mode (ATM) is considered to be the most promising transfer technique because of its efficiency and flexibility.

In ATM networks, the performance bottleneck of the network, which was once the channel transmission speed, is shifted to the processing speed at the network switching nodes and the propagation delay of the channel. This shift is because the high-speed channel increases the ratio of processing time to packet transmission time

and also the ratio of propagation delay to packet transmission time. The increased processing overhead makes it difficult to implement hop-by-hop schemes, which may impose prohibitably high processing at each switching node. The increased propagation delay overhead makes traffic control in ATM a challenge since a large number of packets can be in transit between two ATM switching nodes. Because of these fundamental changes, control schemes developed for traditional networks may not perform efficiently, and thus, new network architectures (congestion control schemes, error control schemes, etc.) are required in ATM networks.

In this dissertation, we first present an extensive survey of various traffic control schemes and network protocols for ATM networks. In this survey, possible traffic control schemes are examined, and problems of those schemes and their possible solutions are presented. Next, we investigate two key research issues in ATM networks (and other types of high-speed networks): the effects of protocol-processing overhead and the efficiency of traffic control schemes.

We first investigate the effects of protocol-processing overhead on the performance of error recovery schemes. Specifically, we investigate the performance tradeoffs between link-by-link and edge-to-edge error recovery schemes. Our results show that for a network with high-speed/low-error-rate channels, an edge-to-edge scheme gives a smaller delay than a link-by-link scheme. We then investigate the effectiveness of a priority packet discarding scheme, a congestion control mechanism suitable for high-speed networks. We derive loss probabilities for each stream and investigate the impact of burstiness of traffic streams on the performance of individual streams.

Chapter 1

Introduction

Due to the increased demand for communication services of all kinds (e.g., voice, data and video), B-ISDN has received increasing attention in the past few years. The key to a successful B-ISDN system is the ability to support a wide variety of traffic sources as well as diverse service and performance requirements. B-ISDN is required to support traffic requiring bandwidth ranging from a few Kbits/sec (e.g., a slow terminal) to several hundred Mbits/sec (e.g., moving image data). Some traffic, such as interactive data and video, is highly bursty; while some traffic, such as large file transfers, is continuous. B-ISDN is also required to meet diverse service and performance requirements of multimedia traffic. Real-time voice, for instance, requires rapid transfer through a network, but a loss of small amounts of voice information is tolerable. In many data applications, real-time delivery is not of primary importance, but high throughput and strict error control are required. Some services, such as real-time video communications, require error-free transmission as well as rapid transfer [IAU87].

B-ISDN should also be able to facilitate expected (as well as unexpected) future services in a practical and easily expandable fashion. Examples of expected future services include high-definition TV (HDTV), broadband videotex, and video/document retrieval services [HM89, HAND89].

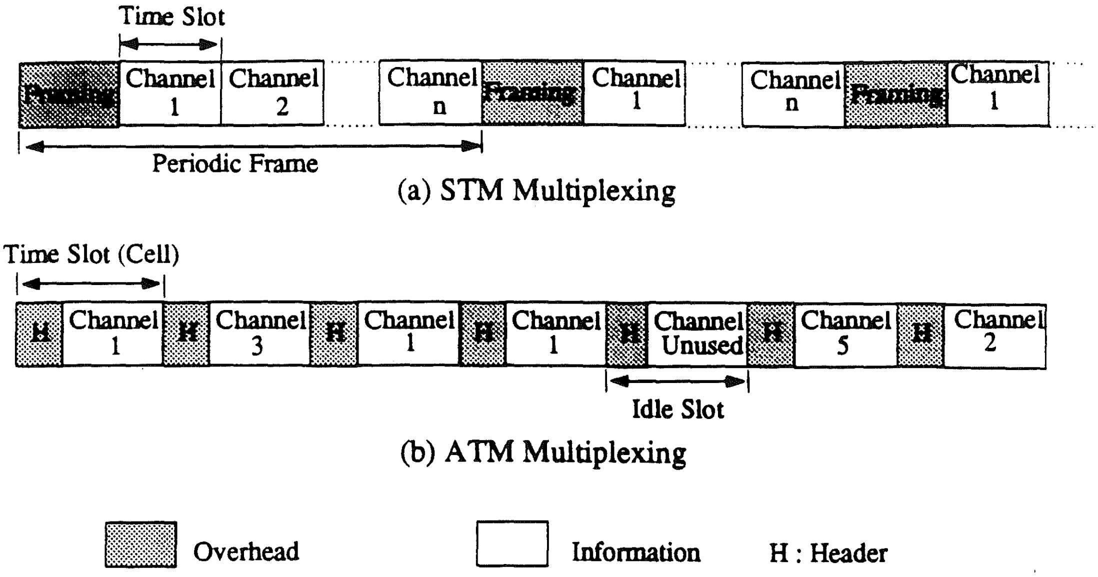


Figure 1.1: STM and ATM Principles

1.1 Asynchronous Transfer Mode (ATM)

To meet the above requirements for a successful B-ISDN, several techniques have been proposed for the switching and multiplexing schemes ("transfer mode"). These schemes include circuit-switching based Synchronous Transfer Mode (STM) and packet-switching based Asynchronous Transfer Mode (ATM).

STM, a circuit switching based technique, was initially considered an appropriate transfer mode for B-ISDN because of its compatibility with existing systems. In STM, bandwidth is organized in a periodic frame, which consists of time slots. See Figure 1.1 (a). A framing slot indicates the start of each frame. As in traditional circuit switching, each slot in an STM frame is assigned to a particular call, and the call is identified by the position of the slot. In STM, slots are assigned based on the peak transfer rate of the call so that the required service quality can be guaranteed even at the peak load. Because of its circuit-like nature, STM is suitable for fixed-rate services; however, STM cannot support bursty traffic efficiently since, in STM, bandwidth is wasted during the period in which information is transported below peak rate.

ATM eliminates the inflexibility and inefficiency found in STM. In ATM, information flow is organized into fixed-size blocks called "cells," each consisting of a header and an information field. Cells are transmitted over a virtual circuit, and routing is performed based on the VCI (Virtual Circuit Identifier) contained in the cell header. The cell transmission time is equal to the slot length, and slots are allocated to a call on demand. See Figure 1.1 (b). The fundamental difference between ATM and STM is that in ATM, slot assignments are not fixed, instead, the time slots are assigned in an asynchronous (demand-based) manner. In ATM, therefore, no bandwidth is consumed unless information is actually being transported.

Between ATM and STM, ATM is considered to be most promising because of its efficiency and flexibility. Because slots are allocated to services on demand, ATM can easily accommodate variable bit rate services. Moreover, in ATM, no bandwidth is consumed unless information is actually being transmitted. ATM can also gain bandwidth efficiency by statistically multiplexing bursty traffic sources. Since bursty traffic does not require continuous allocation of the bandwidth at its peak rate, a large number of bursty traffic sources can share the bandwidth. ATM can also support circuit-oriented and continuous-bit-rate services by allocating bandwidth based on the peak rate (given that sufficient resources are available). Because of these advantages, ATM is considered more suitable for B-ISDN.

1.2 Fundamental Changes

ATM networks are characterized by very high bandwidth channels (e.g., optical fiber) and high-speed switches. The use of high-speed channels and switches has caused significant impact on the design of ATM and other types of high-speed networks. Fist, the ratio of processing time to packet transmission time has increased. For instance, the transmission time of a 1000 bit packet on a 1 Mbits/sec channel is 1 msec, whereas that of the same packet on a 1 Gbits/sec channel is 1 μsec. This reduction in the packet transmission time makes the protocol-processing time at a switching node comparatively large. Therefore, the current network architectures (e.g., X.25, ISO 7 layer architecture), which employ strict error control between adjacent switching nodes, may not be suitable for a high-speed network. They ensure high quality data transport through a network at the expense of increased processing overhead at each switching node.

Second, the ratio of propagation delay to packet transmission time has increased. As an example, consider two adjacent switching nodes, A and B, linked by a 200 Km cable. Assume 1000 bit packets and the typical propagation delay time of 5 μsec per 1 Km of a cable. Consider the following scenario. Assume a 1 Mbits/sec channel speed. Then, the packet transmission time becomes 1 msec. Node A starts transmitting a packet. It takes 1 msec for the electric signal to propagate to node B. Thus, when the first bit of the packet reaches B, A is transmitting the last bit of the same packet. Let's replace the channel with a 1 Gbits/sec fiber optic cable. The packet transmission time reduces to 1 μsec, while the propagation delay time remains the same. Again, A starts transmitting a packet. This time, when the first bit of the packet arrives at B, A is transmitting the 1,000-th packet. 1,000 packets are already on the channel propagating towards B. This example shows that in high-speed networks such as ATM networks, overhead due to propagation delay time becomes significant. Thus traffic control schemes, such as those which adjust A's input rate based on feedback from B, may not work in ATM networks.

As clearly shown in the above examples, having high-speed channels and switches changes the network situation dramatically; in high-speed networks, the performance bottleneck of the network, which was once the channel transmission speed, is shifted to the processing speed at the network switching node and the propagation delay of the channel. Therefore, some of the control schemes developed for existing networks may no longer be applicable in high-speed networks, and new network architectures (error control schemes, congestion control schemes, etc.) are required.

1.3 Major Contributions

Considering the fundamental changes discussed in the above section, there are two important research issues in high-speed networks: (1) the impact of the protocol-processing overhead on the network performance, and (2) effective control schemes for multimedia traffic. This dissertation focuses on these two key research issues, and the major contributions of this dissertation include the following:

1. We presented an extensive survey of the state of the art ATM network technology. In particular, we surveyed various mathematical models for multimedia traffic sources, congestion control schemes, priority schemes to support multiple traffic classes, and the CCITT ATM standards. Through this survey, problems of various traffic control schemes were identified and possible solutions were presented.

2. We investigated the effects of protocol-processing overhead on the performance of error recovery schemes in high-speed network environments. ATM networks attempt to minimize the bottleneck at the switching nodes by simplifying the lower link-to-link layer protocols. Specifically, the error recovery protocols of the lower link-to-link layers are pushed to the higher edge-to-edge (end-to-end) layers. Selecting edge-to-edge error recovery techniques moves the error correction function to a higher level of the protocol and simplifies the protocol processing required at the network switches. Current schemes for ATM networks employ this simplified network protocol, but the validity of this approach had not been established. In this research, we investigated the edge-to-edge scheme and compared its performance against the performance of the link-by-link scheme, a scheme commonly used in existing networks. We showed that for a network with high-speed/low-error-rate channels, the edge-to-edge scheme gives a smaller packet transmission delay than the link-by-link scheme for both Go-Back-N and Selective-Repeat retransmission procedures, while keeping the packet loss probability sufficiently small. This verified the validity of using a simplified error recovery protocol in high-speed networks.

3. We developed an analytical framework for a priority packet discarding scheme, a traffic control scheme suitable for high-speed networks, and investigated the efficiency of this scheme. In our framework, unlike previous works, we assumed heterogeneous traffic streams to explicitly model multimedia traffic sources and used Markov Modulated Arrival Processes to model bursty nature of the traffic. The loss probability of each arrival stream is obtained in both continuous-time and discrete-time cases following a new stochastic integral approach. A new characterization of an arrival stream, which we refer to as self-loss, is introduced

to qualitatively predict the effects of multiplexing bursty streams with non-bursty streams. The important contribution of this work is twofold – the study of the impact of traffic heterogeneity on the performance of an individual traffic stream and the study of the effectiveness of the priority packet discarding.

In summary, in this dissertation, we attacked the key research issues in high-speed networks and provided significant insights into optimal design of high-speed networks.

1.4 Outline

The organization of this dissertation is as follows: in Chapter 2, a number of important research topics in ATM networks are surveyed. The topics covered include mathematical modeling of various types of traffic sources, congestion control schemes for ATM networks, and priority schemes to support multiple classes of traffic. Standard activity for ATM networks are also presented.

In Chapter 3, the effects of protocol-processing overhead on the performance of error recovery schemes are investigated in high-speed network environments. Mathematical models are built for various error control techniques, and queueing analyses are presented considering the effect of protocol-processing overhead. Simulations are also carried out to verify the validity of some of the assumptions made.

In Chapter 4, a statistical multiplexer where heterogeneous traffic streams are multiplexed is analyzed, and this analysis is extended to accommodate a priority packet discarding scheme, a congestion control mechanism suitable for high-speed networks. We consider the class of Markov Modulated Arrival (MMA) Processes both in continuous time and in discrete time and present an exact analysis of individual packet loss for MMA streams. Our analysis is then applied to investigate the effects of individual traffic characteristics and traffic mix on the individual packet loss probabilities. The effectiveness of priority packet discarding is also investigated through numerical examples.

In Chapter 5, concluding remarks are made, and possible future research problems are discussed.

Chapter 2

Survey of Traffic Control Schemes and Protocols in ATM Networks

Due to the fundamental changes explained in Section 1.2 (i.e., the increased overhead of protocol-processing time and propagation delay), new network architectures are required for ATM networks.

In this chapter, we present an extensive survey of various traffic control schemes and network protocols for ATM networks. The organization of this chapter is as follows: in Section 2.1, various mathematical models proposed for data, voice and video are surveyed. Accurate modeling of traffic sources is important to evaluate the performance of ATM networks. In Section 2.2, congestion control schemes suitable for ATM networks are examined. In Section 2.3, various priority schemes proposed to support multiple service classes are discussed. In Section 2.4, ATM standardization activities are presented. In Section 2.5, a summary of this chapter is given.

2.1 Modeling of Multimedia Traffic Sources

As mentioned in Chapter 1, ATM networks must support various communication services, such as data, voice and video, each having different traffic characteristics. To evaluate the performance of such networks, accurate source modeling is required. The purpose of this section is to examine several traffic models proposed for data, voice and video sources. The various mathematical models described below have been examined against actual measured data, and their accuracy has been validated.

2.1.1 Input Traffic Models for Data Sources

It is well known that generation of data from a single data source is well characterized by a Poisson arrival process (continuous time case) or by a geometric interarrival process (discrete time case). For interactive data transmission, a single cell may be generated at a time. For a bulk data transmission, such as a file transfer, a large number of cells may be generated at a time (batch arrivals).

In existing packet networks, packets could be either of variable or constant length. In ATM networks, however, the cell size is fixed. Furthermore, because the size of a cell is relatively short compared to the length of a packet in existing networks, multiple cells may be created from one data packet.

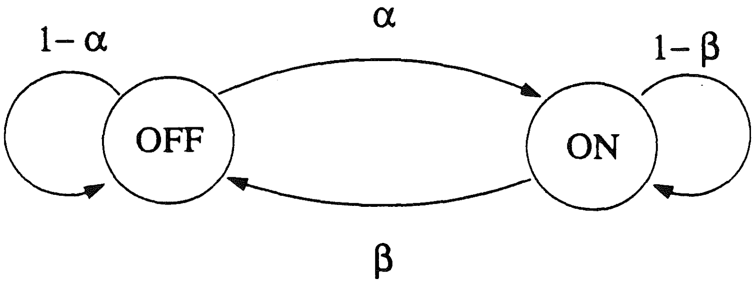


Figure 2.1: IPP Model

2.1.2 Input Traffic Models for Voice Sources

An arrival process of cells from a voice source (and a video source) is fairly complex due to the strong correlation among arrivals. In this subsection, input traffic models proposed for a voice source are examined.

The arrival process of new voice calls and the distribution of their durations can be characterized by a Poisson process and an exponential distribution, respectively. Within a call, talkspurts and silent periods alternate. During talkspurts, voice cells are generated periodically; during silent periods, no cells are generated. The correlated generation of voice cells within a call can be modeled by an Interrupted Poisson Process (IPP) [KS89, MOSM89, LI89b, DL86, IDE88]. In an IPP model, each voice source is characterized by ON (corresponding to talkspurt) and OFF (corresponding to silence duration) periods, which appear in turn. The transition from ON to OFF occurs with the probability β, and the transition from OFF to ON occurs with the probability α. In a discrete time case, ON and OFF periods are geometrically distributed with the mean $1/\beta$ and $1/\alpha$, respectively. Cells are generated during the ON period according to a Bernoulli distribution with the rate λ; no cell is generated during the OFF period. See Figure 2.1. (The continuous time analog is an exponential distribution using a Poisson process.)

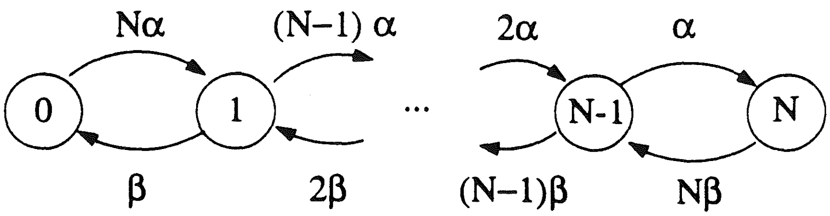


Figure 2.2: Birth-Death Model for the Number of Active Voice Sources

When N independent voice sources are multiplexed, aggregated cell arrivals are governed by the number of voice sources in the ON state. Assuming a discrete time system, the probability P_n that n out of N voice sources are in the ON state (n voice cell arrivals in a slot) is given by

$$P_n = \binom{N}{n}(\frac{\alpha}{\alpha+\beta})^n(\frac{\beta}{\alpha+\beta})^{N-n} \quad for\ 0 \le n \le N. \tag{2.1}$$

The continuous time analog represents the number of voice sources in the ON state as a birth-death process with birth rate $\lambda(n)$ and death rate $\mu(n)$, where

$$\lambda(n) = (N-n)\alpha, \qquad \mu(n) = n\beta, \quad for\ 0 \le n \le N. \tag{2.2}$$

See Figure 2.2. For this continuous time case, the probability P_n that n out of N voice sources are in the ON state is also given by Eq.(2.1) [LI89b].

Another common approach for modeling aggregate arrivals from N voice sources is to use a two-state Markov Modulated Poisson Process (MMPP) [HL86, LM89]. The MMPP is a doubly stochastic Poisson process where the rate process is determined by the state of a continuous-time Markov chain [HL86]. In the two-state MMPP model, an aggregate arrival process is characterized by two alternating states. It is usually assumed that the duration of each state follows a geometrical (discrete time case) or an exponential (continuous time case) distribution, and cell arrivals in each state follow a Bernoulli (or a Poisson) distribution with different rates. Therefore, four parameters are necessary to describe an MMPP: the mean duration of each state and the arrival rate in each state. Note that an IPP, a process used to describe a single voice source, is a special case of the MMPP in which no cell arrives during an OFF period.

To determine the values of these four parameters, the following MMPP statistical characteristics are matched with the measured data [HL86]:

1. The mean arrival rate
2. The variance-to-mean ratio of the number of arrivals in a time interval $(0, t_1)$
3. The long term variance-to-mean ratio of the number of arrivals
4. The third moment of the number of arrivals in $(0, t_2)$

Note that the analytical models described in subsections 2.1.1 and 2.1.2 can model only Constant Bit Rate traffic. Analytical models which can adequately model Variable Bit Rate traffic are not yet available.

2.1.3 Input Traffic Models for Video Sources

Video traffic requires large bandwidth. For instance, in TV applications a frame of 512×512 resolution is transmitted every 1/30 second, generating $512 \times 512 \times 8 \times 30$ bits per second (approximately 63 Mbits/sec), if a simple PCM coding scheme is used. Therefore, video sources are usually compressed by using an interframe variable-rate coding scheme which encodes only significant differences between successive frames. This introduces a strong correlation among cell arrivals from successive frames.

Like a voice source, a video source generates correlated cell arrivals; however, its statistical nature is quite different from a voice source. Two types of correlations are evident in the cell generation process of a video source: short-term correlation and long-term correlation. Short-term correlation corresponds to uniform activity levels (i.e., small fluctuations in bit rates), and its effects last for a very short period of time (on the order of a few hundred milliseconds). Long-term correlation corresponds to sudden scene changes, which cause a large rate of arrivals, and its effects last for a relatively long period of time (on the order of a few seconds) [SMRA89]. In subsection 2.1.3.1, models which consider only short-term correlation (i.e., models for video sources without scene changes) are examined. In subsection 2.1.3.2, models which consider both short-term and long-term correlation (i.e., models for video sources with scene changes) are examined.

2.1.3.1 Models for Video Sources Without Scene Changes

In this subsection, models proposed for video sources WITHOUT scene changes are examined. These models are applicable to video scenes with relatively uniform activity levels such as videotelephone scenes showing a person talking. Two models have been proposed. The first model approximates a video source by an autoregressive (AR) process [NFO89, MASKR88]. This model describes the cell generation process of a video source quite accurately. However, because of its complexity, queueing analysis based on this model is very complicated and may not be tractable. This model is more suitable for use in simulations. The second model approximates a video source (or video sources) by a discrete-state Markov model [MASKR88]. This model is more tractable in queueing analysis than the first model, and yet describes the cell generation process of a video source (or video sources) well.

Model A: Continuous-State Autoregressive Markov Model [MASKR88]

Here, a single video source is approximated by an autoregressive (AR) process. The definition of an AR process is as follows:

$$\lambda(n) = \sum_{m=1}^{M} a_m \lambda(n-m) + bw(n) \tag{2.3}$$

where $\lambda(n)$ represents the source bit rate during the n-th frame; M is the model order; $w(n)$ is a Gaussian random process; and a_m $(m = 1, 2, \cdots, M)$ and b are coefficients. It is shown that the first-order autoregressive Markov model,

$$\lambda(n) = a_1 \lambda(n-1) + bw(n) \tag{2.4}$$

is sufficient for engineering purposes. Assuming that $w(n)$ has the mean η and the variance 1, and that $|a_1|$ is less than 1, the values of coefficients a_1 and b are determined by matching the steady-state average $E(\lambda)$ and discrete autocovariance $C(n)$ of the AR process with the measured data. $E(\lambda)$ and $C(n)$ of the AR process in Eq.(2.4) are given by [PAPO84]

$$E(\lambda) = \frac{b}{1-a_1}\eta, \qquad C(n) = \frac{b^2}{1-a_1^2} a_1^n \qquad n \geq 0 \tag{2.5}$$

This model provides a rather accurate approximation of the bit rate of a single video source without scene changes. However, as stated above, analysis of a queueing model with the above arrival process can be very complex and may not be tractable; therefore, this model is suitable for use in simulations.

Model B: Discrete-State, Continuous-Time Markov Process [MASKR88]

The process $\lambda(t)$ describing the bit rate of a video source at time t is a continuous-time, continuous-state process. In this model, process $\lambda(t)$ is sampled at random Poisson time instances and the states are quantized at these points. See Figure 2.3.

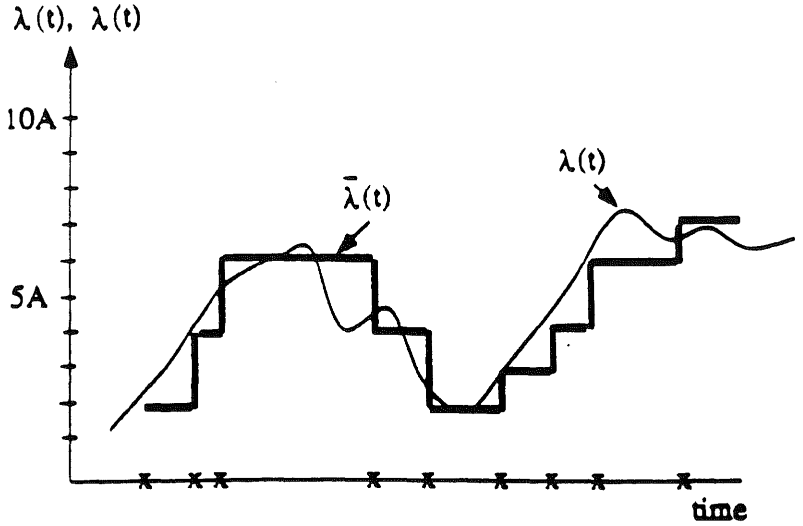


Figure 2.3: Poisson Sampling and Quantization of the Source Rate

In other words, the process $\lambda(t)$ is approximated by a continuous-time process $\overline{\lambda}(t)$ with discrete jumps at random Poisson times. This approximation can be improved by decreasing the quantization step A and increasing the sampling rate.

The state transition diagram for $\overline{\lambda}(t)$ is shown in Figure 2.4. The process $\overline{\lambda}(t)$ can be used to describe a single source, as well as an aggregation of several sources. The aggregated arrival process from N video sources can transit between $M+1$ levels. The label in each state indicates the data rate in that state. (A is a constant). To determine values of the quantization step A and the transition rates α and β, the steady-state mean $E(\overline{\lambda}_N)$, variance $\overline{C}_N(0)$, and autocovariance function $\overline{C}_N(\tau)$ of the

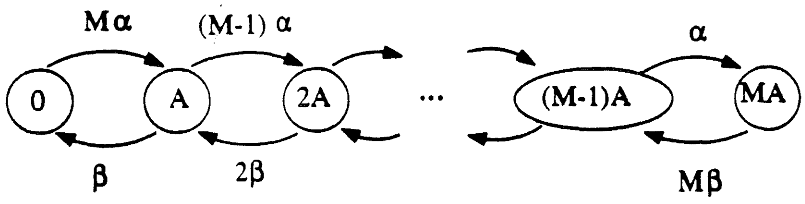


Figure 2.4: State Transition Diagram - Model B

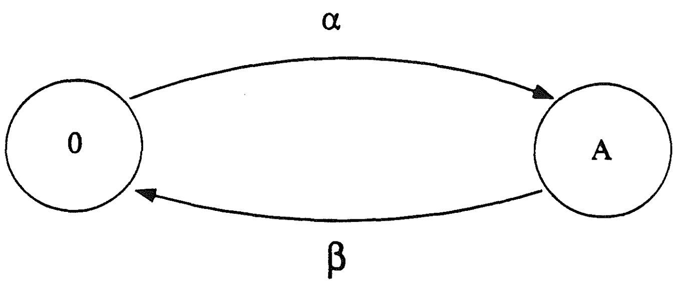


Figure 2.5: Minisource Model

process $\overline{\lambda}(t)$ (describing an aggregate of N independent sources) are matched with the measured data. (τ is a time parameter.) $E(\overline{\lambda}_N)$, $\overline{C}_N(0)$ and $\overline{C}_N(\tau)$ are given by

$$E(\overline{\lambda}_N) = MA\frac{\alpha}{\alpha+\beta}, \quad \overline{C}_N(0) = MA^2\frac{\alpha}{\alpha+\beta}(1-\frac{\alpha}{\alpha+\beta}), \quad \overline{C}_N(\tau) = \overline{C}_N(0)e^{-(\alpha+\beta)\tau} \tag{2.6}$$

The number of quantization levels M is chosen arbitrarily, but it should be large enough to cover all likely bit rates.

The process in Figure 2.4 can be decomposed into a superposition of simpler processes. It can be thought of as a superposition of M independent identical ON-OFF minisources, each being modeled as in Figure 2.5. Each minisource alternates between ON and OFF states. The transition from ON to OFF state occurs with the rate β, and the transition from OFF to ON state occurs with rate α. (Thus, both ON and OFF periods are exponentially distributed.) The data rate of a minisource in the ON state is A; a minisource does not generate bits during the OFF state (data rate is 0). (Note that in Figure 5, a label associated with the state represents the data rate of a minisource in that state.) The state of the aggregated arrival process can thus be represented by the number of minisources which are in the ON state.

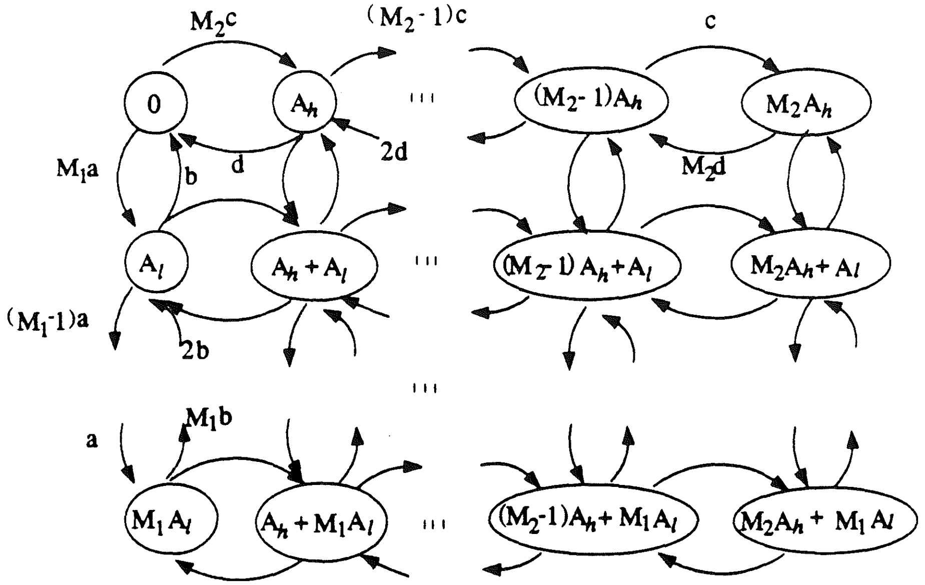


Figure 2.6: State-Transition-Rate Diagram for the Aggregate Source Model (With Scene Changes)

2.1.3.2 Models for Video Sources With Scene Changes

In this subsection, models proposed for video sources *with* scene changes are examined. These models capture both short-term and long-term correlations explained at the beginning of subsection 2.1.3, and thus, these models are suitable to describe a cell generation process from video scenes with sudden changes, such as videotelephone scenes showing changes between listener and talker modes, or scene changes in broadcast TV [SMRA89]. Two models have been proposed: the first model is an extension of Model B explained above; the second model approximates a video source by the discrete-state continuous-time Markov process (Model B) with batch arrivals.

Model C: An Extension of Model B [SMRA89]

The state transition diagram of the cell generation process from an aggregation of N video sources is shown in Figure 2.6. (This process can also be used to describe

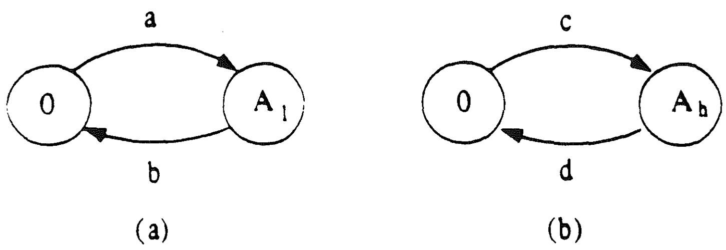


Figure 2.7: Miniprocess Models

a single video source with scene changes.) The label in each state indicates the data rate in that state. There are two basic data rate levels: a high data rate A_h, which represents a sudden scene change, and a low data rate A_l, which represents a uniform activity level. If scene changes do not exist (i.e., if we delete the states which contain a high rate A_h), the process in Figure 2.6 reduces to the one used in Model B. The aggregated process of N video sources can transit between $(M_1 + 1)(M_2 + 1)$ levels, where $M_1 = NM$, $M_2 = N$. Here, M is chosen arbitrarily.

To determine the values of system parameters c and d (the transition probabilities between uniform activity level and high activity level), the fraction of the time spent in the high activity level ($\frac{c}{c+d}$) and the average time spent in the high activity level (1/d) are equated with the actual measured data. To determine the rest of the parameters in the model, i.e., the transition probabilities within the uniform activity level (a and b), and the two basic data rates (A_l and A_h), the first and second order statistics are matched with the actual measured data.

As in Model B, the process described in Figure 2.6 can be decomposed into a superposition of simpler processes. This process can be thought of as a superposition of M_1 independent identical ON-OFF minisources of the type shown in Figure 2.7 (a) and M_2 of the type shown in Figure 2.7 (b). The state of the aggregated arrival process can thus be described as the number of each type of minisource which is in the ON state.

Model D: Discrete-State Continuous-Time Markov Process with Batch Arrivals [YYOK89, YMKM89]

In this model, the cell arrival process from a single video source *with* scene changes is modeled as a discrete-state continuous-time Markov process with batch arrivals. The uniform activity level is represented by a discrete-state continuous-time Markov process as in Model B. This M-state Markov process can be decomposed into M independent identical ON-OFF minisources. Scene changes (high activity levels) are represented by a batch arrival process. The interarrival times between scene changes (between batches) are assumed to be exponentially distributed, and the batch size is assumed to be constant.

2.2 Congestion Control in ATM Networks

In an ATM network, most traffic sources are bursty. A bursty source may generate cells at a near-peak rate for a very short period of time and immediately afterwards it may become inactive, generating no cells. Such a bursty traffic source will not require continuous allocation of bandwidth at its peak rate. Since an ATM network supports a large number of such bursty traffic sources, statistical multiplexing can be used to gain bandwidth efficiency, allowing more traffic sources to share the bandwidth. But notice, if a large number of traffic sources become active simultaneously, severe network congestion can result.

Due to the fundamental changes discussed in Chapter 1, congestion control is a challenge for an ATM network. The increased ratio of processing time to packet transmission time makes the protocol-processing time a bottleneck. In order to avoid such a bottleneck, ATM networks use simplified protocols, pushing most of the link-to-link layer protocols to higher edge-to-edge layers. This makes it difficult to implement link-by-link congestion control schemes. Further, due to the increased ratio of propagation delay to packet transmission time, congestion control schemes based on feedback may not be efficient in ATM networks.

For these reasons, many of the congestion control schemes developed for existing networks may not be applicable to ATM networks. Many of the congestion control schemes developed for existing networks fall in the class of reactive control. Reactive control reacts to the congestion after it happens and tries to bring the degree of network congestion to an acceptable level. However, reactive control is not suitable for use in ATM networks.

A new concept is therefore required for congestion control in an ATM environment. Various congestion control approaches have been proposed for ATM networks, most of which fall in the class of preventive control. Preventive control tries to prevent congestion before it happens. The objective of preventive control is to ensure *a priori* that network traffic will not reach the level which causes unacceptable congestion. In the following, we first explain the reasons that reactive control does not perform well in ATM networks (in subsection 2.2.1), and then examine various preventive control schemes proposed for ATM networks (in subsection 2.2.2).

2.2.1 Reactive Control

At the onset of congestion, reactive control instructs the source nodes to throttle their traffic flow by giving feedback to them. A major problem with reactive control in high-speed networks is slow feedback. As mentioned previously, the effects of high-speed channels make the overhead due to propagation delay significant; therefore, by the time that feedback reaches the source nodes and the control is triggered, it may be too late to react effectively.

There is a possible improvement technique to overcome the difficulty caused by slow feedback. If reactive control is performed between network users and the edge of the network as in [GL89], the effect of propagation delay may not be significant since the distance feedback information propagates is short. However, this limits the reactive control to the edge of the network.

Because of the problem discussed above, reactive flow control, in general, may not be effective in an ATM environment. Preventive control, however, tries to overcome this problem with reactive control and controls congestion more effectively in ATM networks. Preventive control schemes are examined in the next subsection.

2.2.2 Preventive Control

Unlike reactive control where control is invoked upon the detection of congestion, preventive control does not wait until congestion actually occurs, but rather tries to prevent the network from reaching an unacceptable level of congestion. The most common and effective approach is to control traffic flow at entry points to the network (i.e., at the access nodes). This approach is especially effective in ATM networks because of its connection-oriented transport. With connection-oriented transport, a decision to admit new traffic can be made based on knowledge of the state of the route which the traffic would follow [WRR88].

Preventive control for ATM can be performed in two ways: admission control and bandwidth enforcement. Admission control determines whether to accept or reject a new connection at the time of call set-up. This decision is based on traffic characteristics of the new connection and the current network load. The bandwidth enforcement monitors individual connections to insure that the actual traffic flow conforms with that reported at call establishment. In subsections 2.2.2.1 and 2.2.2.2, admission control and bandwidth enforcement are discussed in detail.

Holtzman has proposed a new and very different approach to preventive control [HOLT89] and has applied his approach to admission control. In admission control, the decision to accept a new connection is made based on the predicted network performance. If there is some uncertainty in the parameter values of the incoming traffic, the network may underestimate the impact of accepting a new call and congestion may result. Holtzman's approach tries to prevent the network congestion by taking uncertainties in traffic parameter values into account. Holtzman's approach is described in subsection 2.2.2.3.

2.2.2.1 Admission Control

Admission control decides whether to accept or reject a new connection based on whether the required performance can be maintained. When a new connection is requested, the network examines its service requirements (e.g, acceptable cell transmission delay and loss probability) and traffic characteristics (e.g., peak rate, average rate, etc.). The network then examines the current load and decides whether or not to accept the new connection.

Three major research issues in admission control are:

- What traffic parameters (traffic descriptors) are required to accurately predict network performance?
- What criteria should the network use to decide whether or not to accept a new connection?
- How does network performance depend on various traffic parameters?

In the following, these three issues are discussed.

Traffic Descriptors

When a new connection is requested, the network needs to know the traffic characteristics of the new connection in order to accurately predict its ability to maintain a certain performance level. A set of traffic descriptors given from a user to a network should include sufficient parameters so that the network can accurately determine the user's traffic characteristics. However, for simplicity's sake a set of traffic descriptors should include the fewest possible parameters.

The peak bit rate, the average bit rate, and a measure of burstiness are the most commonly used parameters for traffic descriptors. Among them, "burstiness" is the most important parameter, especially in an ATM network where most traffic sources are highly bursty. Burstiness is a parameter which describes how densely or sparsely cell arrivals occur. It is well known that burstiness plays a critical role in determining network performance ; however, consensus is yet to be reached concerning an appropriate way to describe the burstiness of a traffic source. Possible definitions of burstiness proposed include:

1. The ratio of peak bit rate to average bit rate [DJ88, KM84, CHOI89, GRF89]

2. The average burst length, i.e., the mean duration of the time interval during which the traffic source transmits at the peak rate [HW89]
3. Burst factor defined as the average number of bits accumulated in a buffer during a burst, namely, (peak bit rate - average service bit rate) × average burst length [AKHT87]
4. Cell jitter ratio defined as the variance-to-mean ratio of the cell interarrival times, namely, Var[cell interarrival times]/E[cell interarrival times] [HA87]
5. The squared coefficient of variation of the interarrival times, namely, Var[cell interarrival times]/E^2[cell interarrival times] [SW86]
6. Peakedness defined as the variance-to-mean ratio of the number of busy servers in a fictitious infinite server group [ECKB83]

Deciding the best way to describe the burstiness is a very difficult task which needs to be studied further. We believe that the burst length should somehow be taken into account since it significantly affects the performance. In [MOSM89, DJ88, HW89, AKHT87, JMDS90], it is shown that the longer the burst length, the worse the network performance becomes; namely, the cell loss probability becomes larger and the cell transmission delay becomes longer. The effect of the average burst length is also examined in [GRF89]. It is shown that with longer bursts, statistical multiplexing becomes less effective, and thus, fewer active sources can be supported for a given amount of bandwidth. The authors also believe that more than one parameter may be necessary to describe burstiness.

In [NKN89], a new traffic descriptor is proposed. In this paper, the difficulty of using the peak bit rate or the average bit rate as a traffic descriptor uniformly across the different types of traffic is realized. If the peak bit rate is used regardless of the type of traffic, a large portion of bandwidth will be wasted, especially when the network traffic is bursty. On the other hand, if the average bit rate is used regardless of the type of traffic, the continuous-bit-oriented (CBO) traffic will suffer severe performance degradation. In [NKN89], a new bit rate, called the effective bit rate, is proposed. An effective bit rate is defined as a fraction of the peak bit rate, namely, effective bit rate = (peak bit rate) × a, where a is a constant. The value of a is determined based on the traffic characteristics of the source. By changing the value of a, we can improve the network resource utilization. Further study is required to determine an appropriate value of a for different types of traffic.

Decision Criteria

The cell transmission delays and the cell loss probabilities, because they are good indications of the degree of network congestion, are the most commonly used decision criteria in admission control. When transmission delays and cell loss probabilities are applied in admission control, their long-term-time-averaged values have been used in the past [DJ88, CHOI89, HW89, JMDS90, NKN89, EKIK89]. Using a long-term-time-averaged value, however, may not be sufficient in an ATM network because here the network traffic can change rapidly and dynamically, forcing the network to move from one degree of congestion to another. Figure 2.8 [KS89] sketches how the cell loss probability changes in an ATM network as a function of time. In this figure, the number of active calls jumps from a at time t_0, to b at time t_1, and to c at time t_2. At time t_3, the number of active calls decreases to b. The solid curve in the figure shows the time-dependent behavior of the cell loss probability. For instance, when the number of active calls increases to b at time t_1, the network responds to the change and starts losing a large number of cells; gradually, the network goes up to the next level of congestion and reaches the value of the cell loss probability in steady state $P_{loss}(b)$. When another increase occurs at time t_2, the network responds again, gradually reaching the steady state, and so on. When the network traffic is highly bursty and

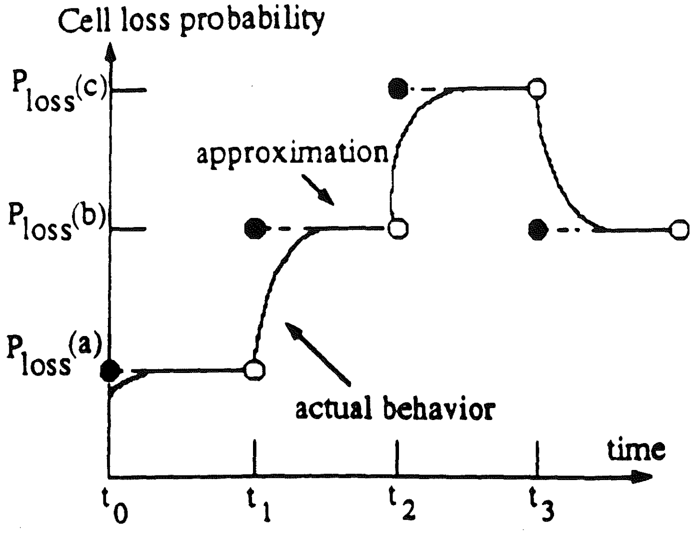


Figure 2.8: Time Dependent Behavior of Cell Loss Probability

changes dynamically, temporal network congestion can occur, and it is possible that a large number of cells are lost during congestion periods, even when the long-term-time-averaged value of loss rate is kept small. In voice communication, for example, this burst loss of voice cells may cause noticeable performance degradation (clicks) at a destination user. Therefore, some decision criteria which take the temporal behavior of the network into account may be needed.

In [KS89], an instantaneous cell loss probability is proposed and used as a decision criterion to consider the temporal behavior of a network. An instantaneous cell loss probability is a time-dependent cell loss probability (function of slot position or time), not the value averaged over a long period of time. The solid curve in Figure 2.8 shows the instantaneous cell loss probability. In [KS89], the instantaneous cell loss probability is approximated by its steady state value (dashed lines in Figure 2.8), and an approximate analysis is developed. A new connection is accepted by the network only when the instantaneous cell loss rate is kept below a threshold value at each switching node for longer than a predetermined percentage of time.

In [KS89], the ineffectiveness of using the long-term-time averaged cell loss probability as a decision criterion is demonstrated through numerical examples using realistic parameter values. It is shown that network congestion can last for a length of time on the order of a hundred milliseconds even when the long-term-time-averaged cell loss probability is kept small. In voice conversation, this congestion period is comparable to a burst (talkspurt) length, and thus, a whole talkspurt can be lost during congestion. It is also shown that this burst cell loss can be avoided by using the instantaneous cell loss probability as a decision criterion in admission control.

In [LI89b], the insufficiency of measuring only the long-term-time averaged cell loss probability is discussed further, and the temporal behavior of voice cell loss probability is studied. Under the realistic parameter values, it is found that the cell loss rate changes slowly and remains at zero most of the time. However, once congestion occurs and the cell loss probability becomes large, the cell loss probability may remain large for a long period, causing voice distortion perceptible at the receiver. It is shown that the average cell loss probability within a blocking period (i.e., the time period during which the buffer is full, and thus, cells are blocked) is much larger than the long-term-time averaged cell loss rate. Therefore, the long-term-time averaged cell loss probability does not reflect the temporal behavior of voice cell loss, and it is not sufficient to measure voice distortion incurred.

Effects of Traffic parameters on ATM Network Performance

One of the important research issues in admission control is to investigate the effect of various traffic parameters on network performance.

In [KS89, DJ88, HW89, AKHT87, JMDS90], the effects of statistical multiplexing of bursty sources in an ATM network are investigated. They investigate how the performance (the cell loss probability and the average delay time) varies as a function of various parameters, such as the number of sources, the peak bit rate, and the burstiness of the sources. Some of the common observations made in these papers follow:

- The average burst length is a very important parameter. As the average burst length increases, the performance degrades, i.e., the cell loss probability and delay time increase significantly [DJ88, HW89, AKHT87, JMDS90].
- As the peak rate of each source is increased, the cell loss probability increases [HW89, AKHT87]. This should be intuitively clear.
- In the case where homogeneous sources are multiplexed, if the offered load (i.e., the number of sources × mean bit rate of each source) is kept constant, the cell loss probability decreases as the number of sources multiplexed increases. The reason for this is that when the number of sources multiplexed increases (keeping the offered load constant), the mean bit rate of each source decreases. The mean bit rate is a product of peak bit rate and the fraction of time in which a source is in the active-state (i.e., the state in which a source is transmitting at the peak rate). Therefore, the reduction in the mean bit rate means the reduction in either the peak bit rate or the burst length (or both). In either case, the cell loss probability decreases [DJ88, AKHT87].
- In the case where heterogeneous sources are multiplexed, high-bit-rate sources dominate the performance; an increase in high-bit-rate traffic causes more significant increases in the cell loss probability than does an increase in low-bit-rate traffic [JMDS90]. A similar observation is made in the case where homogeneous

sources are multiplexed; when high-bit-rate sources are multiplexed, the fluctuation in the cell loss is larger than when low-bit-rate sources are multiplexed [KS89]. This is due to the fact that because of the high bit rate, the number of traffic sources which can be multiplexed on one link is rather limited and not large enough to smooth out the bursty nature of each call.

- The cell loss probability decreases as the offered load decreases [DJ88, JMDS90]. Thus, a very efficient way to lower the cell loss probability is to decrease the offered load by providing larger bandwidth. This is only possible, however, if one can assume that bandwidth is negligibly cheap.

In [GRF89], the effects of traffic parameters on the network performance are investigated and a method is proposed to calculate the bandwidth required to satisfy a given performance requirement. Two different cases are considered: the case where homogeneous traffic sources are multiplexed and the case where heterogeneous sources are multiplexed. In both cases, the peak bit rate (Bp), a measure of burstiness (b) defined as the peak-to-mean bit rate ratio ($\frac{Bp}{Bm}$, where B_m is the mean bit rate), and the mean number of cells (L) generated from a burst are used as traffic descriptors. In the following, we summarize the bandwidth assignment rule proposed for the homogeneous traffic case.

In the case where homogeneous traffic sources are multiplexed, the bandwidth required to satisfy a given cell loss requirement is calculated by

$$W = n\frac{Bp}{b}R(b, n, L) \tag{2.7}$$

where n is the number of active traffic sources; $n\frac{Bp}{b}(= nB_m)$ is the offered traffic; and $R(b, n, L)$ is a coefficient whose value depends on the triplet (b, n, L). $R(b, n, L)$ is called an expansion factor, and its value is obtained by performing a single simulation for each triplet (b, n, L) for a given cell loss requirement. In this paper, a cell loss requirement of 10^{-5} is assumed. This cell loss probability is rather large to be used in a real system; this cell loss probability is used because the simulation run time prohibits the choice of a more realistic cell loss probability of 10^{-9}.

Using Eq.(2.7), if the offered traffic ($n\frac{Bp}{b}$) and the expansion factor $R(b, n, L)$ are given, required bandwidth W can be determined. The remaining question is whether the expansion factor is a function of a triplet (b, n, L), or a function of a quadruplet (Bp, Bm, n, L). In [GRF89], it is claimed that a triplet (b, n, L) is sufficient to determine the expansion factor, and it is supported by examining simulation results for two cases: $Bp = 10$ Mbits/sec or 2 Mbits/sec. To determine that this approach is truly valid, more cases should be examined.

The approach proposed in this paper considers the burstiness of the traffic and uses the peak-to-mean bit rate ratio ($b = \frac{Bp}{Bm}$) and the mean number of cells generated in a burst (L) to determine required bandwidth (W). Even though the approach of using $R(b, n, L)$ to calculate required bandwidth is simpler than the approach in which the quadruplet (Bp, Bm, n, L) is used, it has the following problem. To implement this approach, the values of $R(b, n, L)$ need to be precomputed through the simulation and stored in each node. Therefore, the number of possible combinations of (b, n, L) needs to be tractably small. This may limit the size of the network to which this approach can apply.

2.2.2.2 Bandwidth Enforcement

Since users may deliberately exceed the traffic volume declared at the call set up (i.e., values of their traffic descriptors) and thus easily overload the network, admission control alone is not sufficient. After a connection is accepted, traffic flow of the connection must be monitored to insure that the actual traffic flow conforms

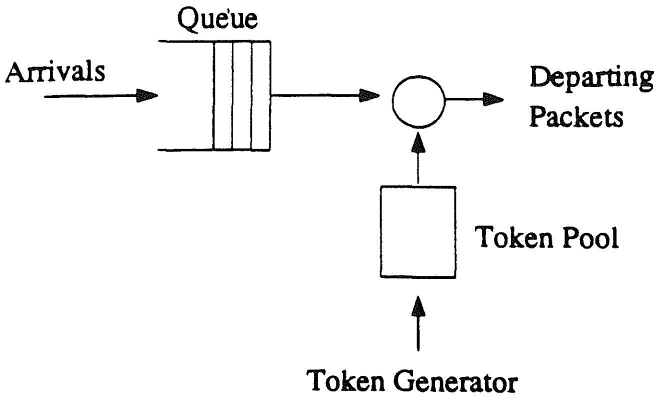


Figure 2.9: Queueing Model for a Leaky Bucket Method

with that specified at call establishment. For this purpose, the bandwidth enforcement mechanism is implemented at the edges of the network. Once a violation is detected, the traffic flow is enforced by discarding and/or buffering violating cells.

A Leaky Bucket method [AKHT87, TURN86, BLM87, CG88] is one of the typical bandwidth enforcement mechanisms used for ATM networks; this method can enforce the average bandwidth and the burst factor of a traffic source. One possible implementation of a Leaky Bucket method is to control the traffic flow by means of tokens. A queueing model for the Leaky Bucket method is illustrated in Figure 2.9 [SLCG89]. An arriving cell first enters a queue. If the queue is full, cells are simply discarded. To enter the network, a cell must first obtain a token from a token-pool; if there is no token, a cell must wait in the queue until a new token is generated. Tokens are generated at a fixed rate corresponding to the average rate of the connection. If the number of tokens in the token pool exceeds some predefined threshold value, the process of token generation stops. This threshold value corresponds to the burstiness of the transmission; the larger the threshold value, the bigger the burstiness. This method enforces the average input rate while allowing for a certain degree of burstiness. The Leaky Bucket method can also enforce the peak bandwidth by generating tokens at the rate corresponding to the peak rate.

In the original Leaky Bucket method proposed in [TURN86], the input buffer is not provided. In [CG88], the input buffer is suggested to provide better control of the trade-off between the cell waiting times and the cell loss probabilities. In an extreme case, where no input buffer is provided, incoming cells do not have to wait in the buffer, but a large number of cells may be lost since all the violating cells are discarded. In the other extreme case (where an infinite input buffer is provided), no incoming cell will be lost, but cells may suffer a long waiting time. By choosing an appropriate input queue size, the trade-off between these two extremes can be controlled. In [SLCG89], an exact analysis of Leaky Bucket methods with and without an input queue is presented, providing the Laplace transforms for the waiting times and the inter-departure times of cells from the system (i.e., inter-departure times of tokens from a token pool). The expected waiting time, the cell loss probability, and the variance of the inter-departure times are also obtained. In this paper, a Poisson process is assumed for the cell arrival process. A Poisson process, however, may not accurately describe bursty traffic found in ATM networks.

In the Leaky Bucket method, violating cells are either discarded or stored in a buffer even when the network load is light, and thus, network resources are wasted. The total network throughput can be improved by using the marking method[1] presented in [GRF89, HW89, ELL89]. In this scheme, violating cells, rather than being discarded, are permitted to enter the network with violation tags in their cell headers. These violating cells are discarded only when they arrive at a congested node. If there are no congested nodes along the routes, the violating cells are transmitted without

[1] Also referred to as a Virtual Leaky Bucket Method.

being discarded. This marking method can easily be implemented using the Leaky Bucket method described above. When the queue length exceeds a threshold, cells are marked as "droppable" instead of being discarded. Through simulations it is shown that by choosing an appropriate threshold value, the marking method can guarantee a performance level required by non-violating cells and at the same time, can improve the network throughput. One possible disadvantage of this marking scheme is that processing time in each node is increased slightly because each node has to distinguish tagged cells from non-violating cells when the node is in a congested state. Each node must also monitor its state to determine if it is in congestion. (For instance, each node may check its queue length to detect the congested state.) However, this extra processing can be done quickly and easily, and the overall merits of the marking method far exceed its slight disadvantages.

An ideal bandwidth enforcement scheme should be able to correctly identify all the violating cells and discard or tag only violating cells. It should also be able to detect violation rapidly once it occurs. However, the bursty nature of the traffic carried in ATM networks makes it difficult to implement such an ideal scheme. When the traffic is bursty, a large number of cells may be generated in a short period of time, yet conform to the traffic descriptor values claimed at the time of call establishment. For instance, the average cell arrival rate can be kept constant if cells do not arrive for a while, even if there is a burst of cell arrivals in a short time period. In this case, none of these cells should be considered violating cells. If a small value is used for a threshold, some of the cells will be falsely identified as violating cells; therefore, a relatively large threshold value must be used to avoid discarding or tagging non-violating cells. However, this large threshold value makes it harder to distinguish truly violating transmissions from temporary burst transmissions; thus, the time required to detect violations is increased. As a result, in an ATM environment it may be more desirable to apply a marking method in order to avoid undesired enforcement actions by the network.

Bandwidth enforcement schemes may also be used with traffic shaping[2]. The purpose of traffic shaping is to throttle cell inputs into a network to avoid the bursty cell transmissions. Burst cell transmissions are avoided, for example, by separating successive ATM cells by idle times. The shaping function could be performed by the access control either at a user-network interface or at a data source by buffering and injecting cells into the network at a slower speed. Since traffic shaping reduces network congestion by suppressing inputs to the network, it may be able to support a greater number of calls than a network without the shaping function. With traffic shaping, the entire transmission of traffic may be unnecessarily slowed since cells are injected into a network at a slower speed even when the network load is light. However, with traffic shaping this degradation in the service quality is achieved in a more graceful way.

2.2.2.3 Coping with Traffic Uncertainties

In the previous subsections, admission control and bandwidth enforcement are examined. In admission control, the network performance is predicted based on the traffic descriptor values provided by the network users, and then a decision is made as to whether a new connection is accepted or not. In bandwidth enforcement, each connection is monitored, and the traffic flow is forced to conform with the traffic descriptor values provided by the network users. However, the exact traffic characteristics may not be available to the network users, and therefore, the values of traffic descriptors provided by the users may involve large uncertainty. In such a case, a

[2] Also referred to as traffic smoothing.

network may underestimate the impact of accepting a new connection and congestion may result.

Very little attention has been paid to the problem of uncertainty in traffic descriptor values. Holtzman addressed this issue in [HOLT89], examining three approaches which were originally proposed in other contexts, and considering their application to the problem of traffic uncertainty in ATM networks. The three approaches examined by Holtzman are the approach using random variables [MW89], the fuzzy set approach [LL89] and the neural net approach to learn about the uncertain environment [HIRA89]. In this subsection, the first approach, which is the most promising and widely applicable, is discussed.

In the first approach discussed by Holtzman, the uncertainty in the traffic descriptor values is quantified by using a random variable for each uncertain parameter in the traffic model. Assume that the cell arrival process to the network is characterized by a point process parameterized by k traffic parameters, $x_1, \cdots, x_k$. Further, assume that the delay incurred by cells through the network in question is a function of th k traffic descriptors and is given by $D(x_1, \cdots, x_k)$. Assume that the performance requirement is given, and it is to keep the delay (mean or percentile) less than a given threshold value D^* (i.e., $D(x_1, \cdots, x_k) < D^*$). Since it is assumed that the delay function $D(x_1, \cdots, x_k)$ is known, we can determine the feasible parameter region Ω to satisfy the performance requirement $D(x_1, \cdots, x_k) < D^*$. Ω is a range of possible values of the traffic descriptors which satisfy a given performance requirement.

Let us denote $\underline{Y}^j = (Y_1^j, \cdots, Y_k^j)$ as a set of random variables which parameterize the arrival stream for the j-th network user. Using $\underline{Y}^j$ ($j = 1, \cdots, n$), the aggregated cell arrival process from n users can be obtained. Let us denote this aggregated arrival process as $\underline{X}^{(n)} = (X_1^{(n)}, \cdots, X_k^{(n)})$. In general, $\underline{X}^{(n)} = f(\underline{Y}^1, \ldots, \underline{Y}^n)$. $\underline{X}^{(n)}$ is a set of random variables which parameterize the cell arrival process from the superposition of n users.

From $\underline{X}^{(n)}$, the number of users n^*, which can be supported by the network satisfying the performance requirement $D(x_1, \cdots, x_k) < D^*$ with high probability, can be determined. n^* is given by $n^* = \max_n\{n : P[\underline{X}^{(n)} \in \Omega] > 1 - \delta\}$, where δ is a predefined tolerance level. (For non-homogeneous superpositions, the traffic mix should be specified.)

In obtaining the aggregated arrival process $\underline{X}^{(n)}$, traffic uncertainties are considered. The process of obtaining $\underline{X}^{(n)}$ can be better illustrated using an example. Assume that the traffic generated by the j-th user is characterized by the mean cell arrival rate and the squared coefficient of variation of the time between cell arrivals. Further, assume that these two parameters have uncertainties. For the traffic generated by the j-th user, assume that:

- The mean cell arrival rate is modeled by a normally distributed random variable Y_1^j, with mean λ_j and variance $\sigma_{\lambda_j}^2$.
- The squared coefficient of variation of the time between cell arrivals is modeled by a normally distributed random variable Y_2^j, with mean c_j^2 and variance $\sigma_{c_j^2}^2$.
- The random variables Y_1^j and Y_2^j are mutually independent.

In the above, the uncertainties in the mean cell arrival rate and the squared coefficient of variation of the time between arrivals are quantified by using random variables Y_1^j and Y_2^j, respectively. Then for the superposed arrival process $\underline{X}^{(n)} = (X_1^{(n)}, X_2^{(n)})$, $X_1^{(n)}$ (the mean arrival rate) and $X_2^{(n)}$ (the squared coefficient of variation of the time between arrivals) need to be calculated. They are calculated using the QNA approximation [WHIT83]. That is,

$$X_1^{(n)} = \sum_{j=1}^{n} Y_1^j, \qquad X_2^{(n)} = W^{(n)}(Z^{(n)} - 1) + 1, \tag{2.8}$$

where

$$Z^{(n)} = \frac{\sum_{j=1}^{n} Y_1^j Y_2^j}{\sum_{j=1}^{n} Y_1^j}, \qquad W^{(n)} = \frac{1}{1 + 4(1 - P^{(n)})(V^{(n)} - 1)}. \tag{2.9}$$

$P^{(n)}$ and $V^{(n)}$ are given by

$$P^{(n)} = s\sum_{i=1}^{n} Y_1^i, \qquad V^{(n)} = \frac{(\sum_{i=1}^{n} Y_1^i)^2}{\sum_{i=1}^{n}(Y_1^i)^2}, \tag{2.10}$$

where s is the mean service time.

Finally, the joint distribution of $X_1^{(n)}$ and $X_2^{(n)}$ needs to be computed. It is found that a bivariate normal distribution is a good approximation. The means and variances of random variables $X_1^{(n)}$ and $X_2^{(n)}$, and the correlation between $X_1^{(n)}$ and $X_2^{(n)}$ are approximated using a Taylor series expansion technique.

Note that although this approach allows uncertainty in parameter values, it must have *a priori* knowledge about the system model (e.g., knowledge about the arrival process and the service process).

2.3 Control of Multiple Traffic Classes

As mentioned earlier, ATM networks must support diversity of service and performance requirements. For instance, real-time voice and video have strict delay requirements, whereas in many data applications, real-time delivery is not a primary concern. Even within delay-sensitive traffic (e.g., voice or video), different traffic streams may have different delay requirements; some data may contain more urgent information than the others. Some traffic (e.g., data) is loss-sensitive and thus must be received without any errors, whereas the inherent structure of speech allows for some loss of information without significant quality degradation. Ideally, uniform control mechanisms should be applied across all the media; however, this is extremely difficult. As an alternative, the notion of multiple traffic classes or GOS (Grade of Service) can be introduced and different control mechanisms can be applied to different traffic classes.

To support multiple classes of traffic in ATM networks, priority mechanisms can be used. Multiple priority levels are provided, and different priority levels are given to different classes of traffic. There are two ways to use priorities; one is to use a priority mechanism as a scheduling method (i.e., queueing discipline). In this way, different delay requirements can be satisfied by scheduling (serving) delay-sensitive or urgent traffic first. The second way is to use a priority scheme to control congestion. In this case, when network congestion occurs, different cell loss requirements can be satisfied by selectively discarding (low priority) cells. In the following subsections, priority schemes are examined in more detail.

2.3.1 Priority Scheme as a Scheduling Method

Various priority schemes can be used as a scheduling method at a switching node in an ATM network. The simplest priority scheme is a static (or fixed) priority scheme. In this scheme, priority is always given to the delay-sensitive class, and the delay-sensitive class is always scheduled for service before the loss-sensitive traffic. This scheme causes relatively high losses for the loss-sensitive traffic while providing relatively low delays for the delay-sensitive traffic.

Since the static priority scheme always schedules the high priority traffic first, if a large portion of the network traffic consists of high priority traffic, the performance for the low priority traffic will be severely degraded. Two dynamic priority schemes, Minimum Laxity Threshold (MLT) and Queue Length Threshold (QLT) [CKT89], try to reduce the performance degradation for the low priority traffic. In these dynamic priority schemes, priority level changes with time.

In the Minimum Laxity Threshold (MLT) scheme, the laxity of a cell is defined as the number of slots remaining before its deadline expires. A cell remains in the queue until either the cell is transmitted or the laxity reaches zero; when the laxity reaches zero, the cell is discarded and considered lost. In this scheme, priority is given to the delay-sensitive traffic if there are any delay-sensitive cells in the queue whose laxity is less than some threshold value; otherwise priority is given to the loss-sensitive traffic. In the Queue Length Threshold (QLT) scheme, priority is given to the loss-sensitive traffic when the number of loss-sensitive cells in the queue exceeds some threshold value; otherwise priority is given to the delay-sensitive traffic.

In both the MLT and QLT schemes, a desired performance level for each of the high and low priority classes can be achieved by choosing an appropriate value for the threshold. The MLT discipline, however, may involve heavy processing at each switching node because the laxity of each real-time cell needs to be updated in every time slot, and each queue needs to be searched to find the minimum laxity cell. Therefore, unless the number of queued cells at each switching node is small, this MLT discipline may not work well in an ATM network, where processing time becomes a bottleneck. In [CKT89], the performance of MLT and QLT disciplines are examined and the analytical models for the two disciplines are developed. Here, little difference in the performance trade-offs is observed in the MLT and the QLT disciplines, and it is concluded that QLT is more practical than MLT due to its simpler implementation.

The above priority schemes differentiate between delay-sensitive and loss-sensitive data in order to meet the performance requirement of each traffic type. As mentioned earlier, even within delay-sensitive traffic, there may be multiple classes, each having a different delay requirement. Head-of-the-Line with Priority Jumps (HOL-PJ) is proposed in [LK88] to satisfy different delay requirements within delay-sensitive traffic. In this scheme, higher priority is given to the class of traffic with stricter delay requirements. It is assumed that each priority class forms its own queue. Within the same priority class, cells are served FCFS, while higher priority queues have non-preemptive priority over lower priority queues. A limit is imposed on the maximum queueing delay of cells within each queue; when the waiting time of a cell exceeds the maximum delay limit, that cell jumps to the end of the next higher priority queue. Thus, the queueing delay of a cell before it joins the highest priority queue is bounded by the sum of the delay limits at all the queues with priorities equal to or higher than the cell's original class. The performance for different classes can be controlled by adjusting the values of the delay limit. A possible disadvantage of this scheme is the processing overhead required for monitoring cells for time-out and moving cells to the next level priority queue. Also, each arriving cell needs to be time-stamped. It is claimed in [LK88] that these tasks are simple, and the processing overhead is relatively small.

There are other priority schemes proposed to satisfy different delay requirements within the delay-sensitive traffic. For instance, refer to [JACK60, JACK61, JACK62, BE89, GOLD77, PTW88] for the Minimum-Laxity-First (MLF) (or Earliest-Due-Date (EDD)) scheme and [CWM89] for the Earliest-Deadline-First (EDF) scheme.

2.3.2 Priority Scheme as a Local Congestion Control Scheme

Priority schemes can be used as local congestion control schemes to satisfy different cell loss requirements of different classes of traffic. For instance, data traffic is loss-sensitive whereas voice traffic can tolerate some loss of information. With a priority scheme, when congestion is detected, priority is given to loss-sensitive traffic (e.g., data) over loss-insensitive traffic (e.g., voice), and cells from lower priority classes are discarded first. This priority scheme recognizes the different cell loss requirements of different classes of traffic. In [YLS88], the impact of discarding voice cells on data traffic is studied. It is shown that the mean waiting time for data can be significantly reduced by discarding voice cells during congestion periods.

In discarding voice information, an improvement can be obtained by selectively discarding voice cells containing less important information. For example, in coded speech, active speech usually carries more important information than background noise during pauses. By discarding cells containing less important information (e.g., background noise), the quaility of the reconstructed voice can be maintained. In [YLS88, PDF89, YSL87, YLS87], priority is given to voice cells containing important information and low priority voice packets are dropped first when congestion occurs. Congestion is controlled locally by selectively discarding voice cells whose loss will have the least effect on the quality of the reconstructed voice signal. It is shown that such a prioritized system is capable of achieving better performance than non-prioritized systems [YLS87].

In the above scheme, a priority level is assigned to each voice cell at the transmitter. The priority level of a cell can be determined by the following methods [YLS87]:

- In the embedded coding method [GOOD80], the encoded information is divided into more significant bits and less significant bits. More significant bits form high priority cells, and less significant bits form low priority cells.
- In the even/odd sample method [JC81], speech samples are identified as either even or odd. Even samples form high priority voice cells, and odd samples form low priority voice cells (or vice versa).
- In the multiple energy level detection method, voice cells (from talkspurts) are classified as "semi-silence" or "active" according to their energy level. (No cells are generated from silent periods.) Priority is placed on "active" voice cells.

Once the cell priority is determined using one of the above methods, low priority cells are discarded at the onset of congestion.

Slightly different techniques to control voice traffic have been proposed. In [SL88, KSB88, DS89, MSZ86], priority is assigned to more important (significant) bits, not to cells. Each cell consists of high priority bits (more significant bits) and low priority bits (less significant bits), and cell size is reduced in response to overload by dropping low priority bits. This technique has a major disadvantage: it requires network nodes to know the internal structure of a voice cell in order to distinguish high priority bits from low priority bits and to manipulate the cell contents [PDF89]. This will increase cell processing at each switching node; thus, this technique may not be suitable for ATM networks. Furthermore, since the cell size is constant in ATM networks, it is not clear how this technique can be applied in ATM networks.

Discarding cells based on the importance of their contents can also be applied to video traffic. If an embedded coding technique[3] [VPV88, KMHY89, GHAN89] is used for the image, coded information is separated into two bit streams: a stream

[3] Also referred to as a layered coding technique or a hierarchical coding technique.

containing essential information and a stream containing picture enhancement information. Cells containing essential information are given higher priority than those containing the picture enhancements. When congestion occurs, only low priority cells are discarded. With this scheme, even when networks become congested the essential parts of coded information are transmitted; thus, it is expected that cell loss will have only a small influence on picture quality [KMHY89].

2.4 Standardization of ATM

In the CCITT Recommendation I.121, a guideline for future B-ISDN standardization, ATM has been accepted as the final transfer mode for B-ISDN [CCITT88a]. According to this Recommendation, information flow in ATM is organized into fixed-size cells, each consisting of a header and an information field. Fixed-size cells are chosen over variable-size units because, based on the state of the existing experimental fast packet switching technology, it is believed that fixed-size cells can be switched more efficiently [MINZ89]. These cells are transmitted over a virtual circuit and cells belonging to the same virtual circuit are identified by the header.

ATM is by definition a connection-oriented technique. This connection-oriented mode minimizes delay variation since cells belonging to the same call follow the same route. It also minimizes the processing required to make routing decisions.

Although agreement has been reached on some aspects of ATM, a number of issues have not yet been resolved. In the CCITT Study Group XVIII meeting held in June 1989, some agreement was reached on the ATM cell format, the underlying transmission system, and the layered architecture for ATM networks. In subsection 2.4.1, the ATM cell format is discussed, and in subsection 2.4.2, standard activities on the underlying transmission structure is discussed. The layered architecture for ATM networks is discussed in subsection 2.4.3.

2.4.1 ATM Cell Format

The size of an ATM cell should be small in order to reduce the degrading effect of the packetization delay at the source. For instance, considerable delay could be introduced during creation of a voice cell if the size of a cell is large. T1S1, a body commissioned by ANSI (American National Standards Institute) to develop ISDN standards for North America, had proposed an ATM cell consisting of a 5-octet header and a 64-octet information field, while ETSI(European Telecommunications Standards Institute), a regional organization that coordinates telecommunications policies in Europe, had proposed an ATM cell consisting of a 4-octet header and a 32-octet information field [MINZ89]. As a compromise, the CCITT has reached an international agreement on an ATM cell consisting of a 5-octet header and a 48-octet information field [CCITT89b, CCITT89a]. The CCITT header formats which will be used at UNI (User-Network Interface) and NNI (Network-Node Interface) are shown in Figure 2.10 [VICK90]. For UNI, the header contains a 4-bit "generic flow control" (GFC) field, a 24-bit label field containing Virtual Path Identifier (VPI) and Virtual Circuit Identifier (VCI) subfields (8 bits for the VPI and 16 bits for the VCI), a 2-bit payload type (PT) field, a 1-bit reserved field, a 1-bit priority (PR) field, and an 8-bit header error check (HEC) field. For NNI, the header does not contain a GFC field, and the extra 4 bits are used for a VPI field.

The GFC field is used to assist the customer premises in controlling the flow of traffic for different qualities of service. The exact procedures for how to use this field are not agreed upon as yet. One candidate for the use of this field is a multiple priority

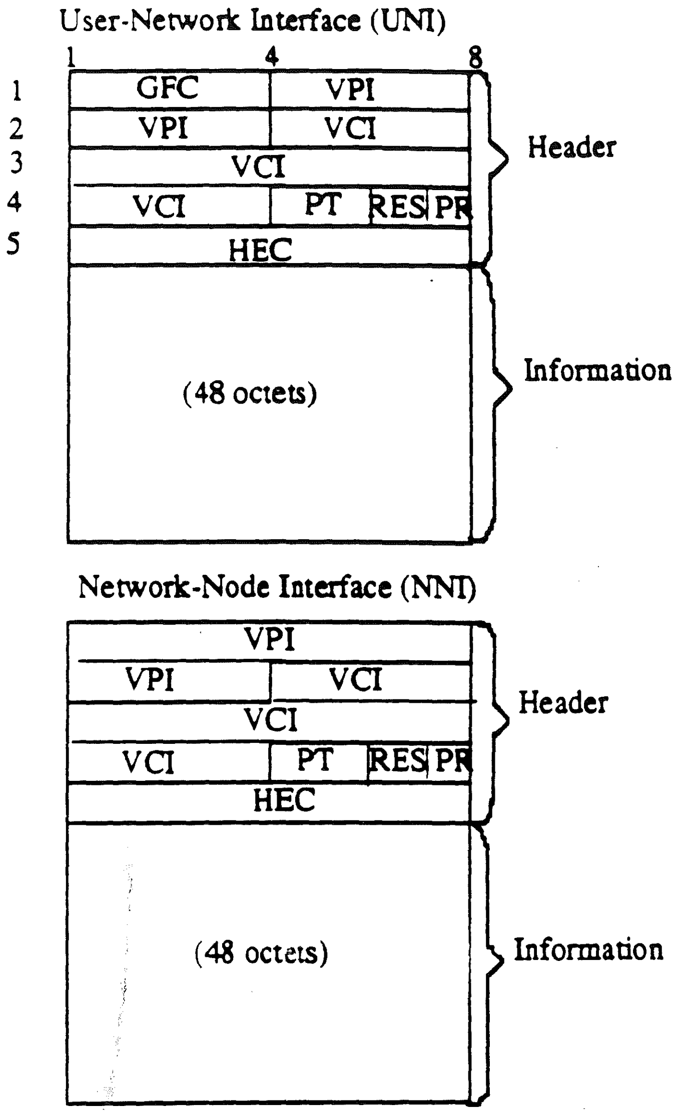


Figure 2.10: ATM Cell Structure

level indicator to control the flow of information in a service-dependent manner. The GFC field appears only at the UNI.

The "Virtual Path" concept [OST88, TST89] is adopted in a label field. The VPI provides an explicit path identification for a cell, while VCI provides an explicit circuit identification for a cell. Basically, a virtual path is a bundle of virtual circuits which is switched as a unit by defining one additional layer of multiplexing on a per-cell basis underneath the VCI. A pre-defined route is provided with each virtual path; thus, it is not necessary to rewrite the routing table at call set-up. Therefore, call-by-call processing at switching nodes is reduced and call set-up delay is decreased. Although the transmission efficiency may decrease because of the label overhead, this effect is negligible since large bandwidth will be available as high capacity optical fibers become more widely used [OST88].

The PT field can be used for maintenance purposes, and it indicates whether the cell contains user information or network maintenance information. This field allows for the insertion of cells on to a virtual channel without impacting the user's data.

The PR field indicates cell loss priority and is used to selectively discard cells when congestion occurs. One possible implementation is to set this field to 0 for the cells which need guaranteed delivery and to set it to 1 for the cells which are droppable. When congestion occurs, cells whose PR field set to 1 are dropped first.

The HEC field provides single-bit error correction or multiple-bit error detection capabilities on the cell header. The polynomial used to generate the header error check value is $X^8 + X^2 + X + 1$. The HEC monitors errors for the entire header. 1 bit in the header is reserved for the future use.

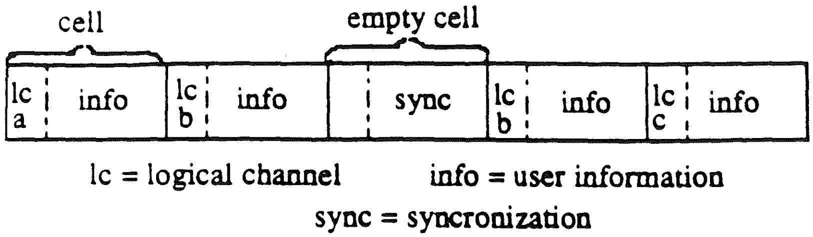


Figure 2.11: ATD Structure

2.4.2 Transmission Structure

T1S1 and ETSI disagrees on the transmission structure underlying the ATM layer. T1S1 favors the SONET (Synchronous Optical Network) approach, whereas ETSI favors the ATD (Asynchronous Time Division) approach [MINZ89]. In the ATD approach, no frame structure is imposed on the UNI, and all of its physical bandwidth is organized as ATM cells; while in the SONET approach, ATM cells are carried within the payload of another framework, such as a SONET frame. The payload is an area used to carry the service or signal being transported.

ATD, a version of ATM originally proposed by France, is a frameless interface carrying no synchronous channels [MINZ87]. See Figure 2.11. It consists solely of cells, and synchronization is maintained by filling empty cells with a special synchronization pattern. The advantage of using ATD as a transmission structure underlying the ATM layer is the simplified interface which results when both transmission and transfer mode functions are based on a common structure. ETSI favors this approach.

The SONET approach will be further discussed in the following subsection.

2.4.2.1 SONET

SONET (Synchronous Optical Network), originally proposed by Bellcore (Bell Communications Research), is a standard optical interface. In this subsection, frame structure and key features of SONET are first presented, and the approach of using SONET as the underlying transmission structure and its advantages and disadvantages are also discussed.

SONET STS-1 Frame Structure

In SONET, there is a basic building block called the synchronous transport signal level 1 (STS-1) frame. The STS-1 frame has a bit rate of 51.84 Mbits/sec and repeats every 125 μ seconds. A 125 μsec frame period supports digital voice signal transport since each byte can represent a 64 Kbits/sec (= 1 byte/125 μsec) DS0 channel.

The STS-1 frame structure is illustrated in Figure 2.12. It consists of 9 rows and 90 columns (9 × 90 bytes), and it is transmitted row by row, from left to right. The STS-1 frame is divided into two areas known as the transport overhead and the Synchronous Payload Envelope (SPE). The transport overhead is used to carry overhead information and the SPE is used to carry the SONET payload. Within the SPE, there is 1 column (= 9 bytes) of path overhead. A path corresponds to a logical connection between source and destination; the functions of path overhead will be discussed later. See Figure 2.13.

The transport overhead consists of 3 columns (= 3 × 9 bytes) and carries overhead bits for connections at the section level (connection between regenerators) and connections at the line level (connection between light-wave terminating equipments). Refer to Figure 2.13 for the concept of section and line. The Section overhead is

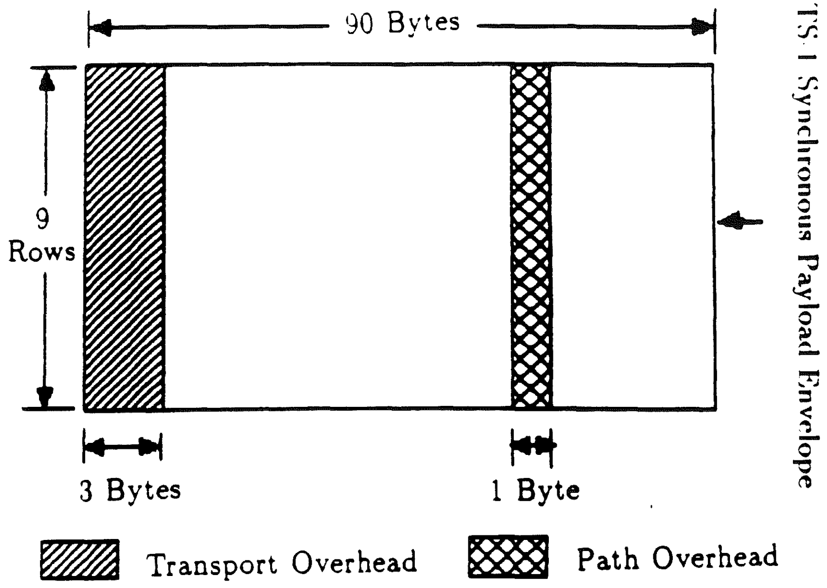


Figure 2.12: STS-1 Frame Structure

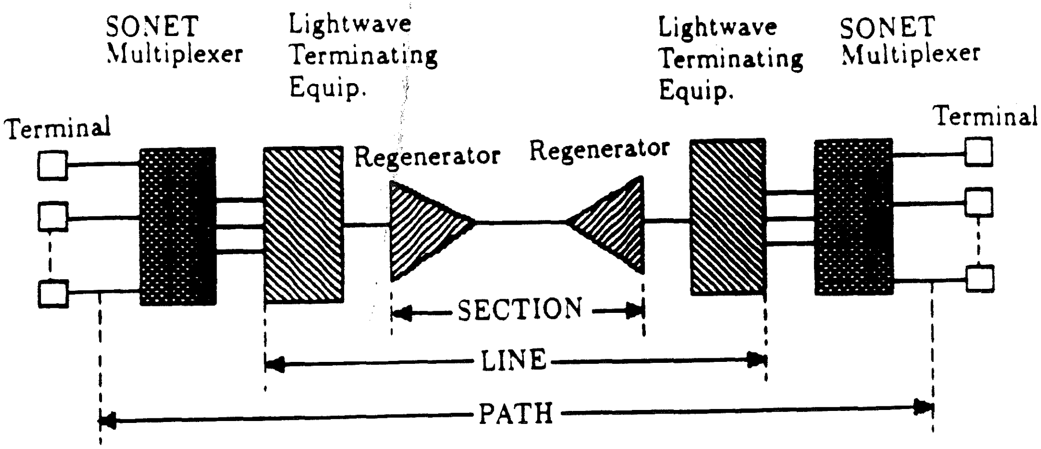


Figure 2.13: Concept of Section, Line and Path

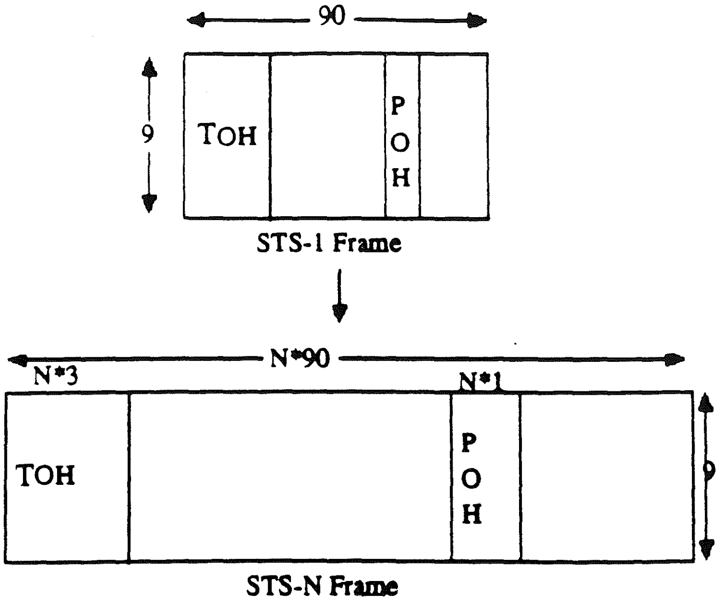
processed at each regenerator. The Line overhead is passed transparently through regenerators and is processed by light-wave terminating equipment. The transport overhead bits and their functions include:

- Framing bytes to show the start of each STS-1 frame
- An STS-1 identification byte
- STS-1 pointer bytes (will be discussed later)
- Parity checks for section and line error monitoring
- Signaling bits for fast, automatic protection switching or redundancy, to make optical lines fault-tolerant
- Local (section) and express (line) orderwire channels for voice communication between elements
- Data communication channels (or embedded operations channels) for alarms, maintenance, control, monitor, administration and other communication needs between section (or line) terminating equipment such as lightwave, cross-connections and digital loop carrier elements
- Extra bytes reserved for the future use

Basically, the transport overhead carries information necessary for secure transmission of the SPE.

The SPE is used to carry SONET payloads including 1 column of path overhead. Path overhead is passed transparently from the point where the STS-1 payload is composed to the point where it is decomposed [HM89]. Some of the important functions of the path overhead are:

- End-to-end payload error monitoring



TOH: Transport Overhead POH: Path Overhead

Figure 2.14: STS-N Frame Structure

- Identification of the type of payload being carried
- Path status indication
- A trace function which allows a user to trace a signal through the network as it goes through different elements

The rest of the SPE is used to carry service or signal being transported. The SONET SPE can be used to carry either ATM-based or STM-based payloads.

Multiplexing of STS-1 Frames

Higher rate SONET signals (STS-N) are obtained by synchronously byte multiplexing N STS-1 frames. The STS-N frame structure is depicted in Figure 2.14. It consists of 9 rows and $N \times 90$ columns. The transport overhead consists of $N \times 3$ columns, and there are $N \times 1$ columns of path overhead. The aggregated bit rate of

Level	Line Rate (Mb/s)
OC-1	51.84
OC-3	155.52
OC-9	466.56
OC-12	622.08
OC-18	933.12
OC-24	1244.16
OC-36	1866.24
OC-48	2488.32

Table 2.1: SONET Digital Interface Rates

an STS-N signal is exactly N times the basic rate of 51.84 Mbits/sec. For example, a STS-3 carries three byte-interleaved STS-1 signals in an aggregate bit rate of 155.52 Mbits/sec. Currently, the only values of N allowed are 1, 3, 9, 12, 18, 24, 36, and 48. The optical signals can be obtained by passing the electrical STS-N signal through an electro-optic conversion. The optical signal of STS-N is called an Optical Carrier Level N (OC-N). The OC-N will have a line rate exactly same as the STS-N. Table 2.1 shows the SONET digital interface rates.

If all the STS-1 signals in the STS-N go to the same destination, they can be concatenated. The STS-N signal that is concatenated is called an STS-Nc, where the letter "c" stands for concatenation. In this format, the payload is treated as a single unit, and thus, only one column of path overhead is needed. This is not the case in the unconcatenated STS-N signals. However, N copies of the section and line overheads are still required in the STS-Nc signal since any intervening transmission equipment expects to see the individual section and line overheads [HM89].

Virtual Tributaries

One of the key features in SONET is payload structures called virtual tributaries (VTs). Virtual tributaries function as separate containers within the STS-1 frame structure and are used to carry a variety of lower rate signals such as DS1, DS1C, DS2 within an STS-1. In order to efficiently accommodate the North American and European digital hierarchy, these containers come with four different sizes: VT1.5, VT2, VT3, and VT6. A VT1.5 can be used to carry a North American DS1 signal (1.544 Mbits/sec), a VT2 for a European CEPT-1 signal (2.048 Mbits/sec), a VT3 for a DS1C signal (3.152 Mbits/sec), and a VT6 for a DS2 signal (6.312 Mbits/sec). A DS3 signal (44.736 Mbits/sec) is carried in an STS-1 SPE.

SONET Pointers

SONET uses payload pointers to allow easy access to the payload. As mentioned earlier, the STS-1 frame is divided into the transport overhead and STS-1 SPE. The payload is easily accessed since the STS-1 payload pointer, contained in the transport overhead, indicates the starting byte location of the STS-1 SPE within the STS-1 frame. The STS-1 payload pointer also avoids the need for the slip buffers and eliminates associated payload corruption and delay. In conventional methods, such as fixed location mapping, slip buffers are needed at the multiplexing equipment interfaces to take care of frequency differences by either repeating or deleting a frame of information. This increases delay and may cause signal impairment due to slipping. The STS-1 payload pointer avoids the need for slip buffers since small frequency variations of the STS-1 payload can be accommodated by adjusting the pointer value. Detailed discussion of pointer operation can be found in [BC89].

SONET can also have VT pointers. The VT pointers follow the same principle as the STS-1 payload pointers, except at the VT level. The VT pointer indicates

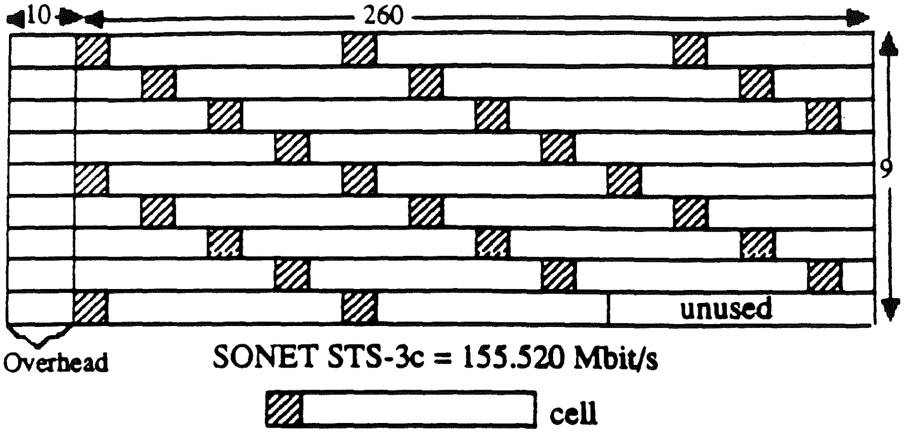


Figure 2.15: ATM within SONET (150 Mbit/s)

the position of the starting byte of the VT SPE within the VT payload structure. The pointer adjustment rules are analogous to that of the STS-1 pointer, and the VT pointer has the same advantages as the STS-1 pointer, i.e., dynamic alignment between the STS-1 SPE and the VT SPE, minimal delay, etc.

ATM within SONET

Figure 2.15 shows ATM carried within SONET. In this figure, it is assumed that the UNI is SONET-based and uses the STS-3c format with a gross bit rate of 155.520 Mbits/sec. B-ISDN proposals using SONET usually use the STS-3c frame [HM89]. SONET overhead is not embedded within the cell structure, and the SONET payload carries ATM cells multiplexed using ATM techniques. T1S1 favors SONET over ATD because of the following reasons [MINZ89]:

- SONET is more compatible with the existing circuit-switched networks than a new structure such as ATD. Furthermore, SONET SPE can be used to carry ATM-based payloads as well as STM-based payloads. Therefore, SONET makes the transition from the existing networks to ATM networks more smoothly than ATD.

- Some specific connection can be circuit switched using a SONET channel. For example, a connection carrying video traffic can be mapped into its own exclusive payload envelope of the SONET STS-3c signal, which can be circuit switched.
- Using SONET synchronous multiplexing capabilities, several ATM streams can be combined to build interfaces with higher bit rates than those supported by the ATM layer. For example, four separate ATM streams, each having bit rate of 155 Mbits/sec (STS-3c), can be combined to build a 622 Mbits/sec (STS-12) interface, even though the ATM layer supports interfaces with bit rate of only 155 Mbits/sec. This may be more cost effective than making the ATM layer to support interfaces with bit rates of 622 Mbits/sec.

A possible disadvantage of using SONET is that existing equipment may not be SONET-compatible. For example, equipment not designed for SONET may not be easily adapted to the VT-1.5 (1.728 Mbits/sec) payload rate of SONET [PRIN89].

CCITT will standardize two physical interfaces to B-ISDN, one based on SONET and the other based on a variation of ATD [CCITT89b]. The UNI interface rate for both of these is set at 155.520 Mbits/sec [CCITT89b].

2.4.3 Layered Architecture for ATM Networks

Significant changes have taken place in ATM networks. ATM networks provide huge bandwidth with low error rates using optical fiber. In such a high-speed network environment, processing time becomes a bottleneck. The conventional OSI 7 layer

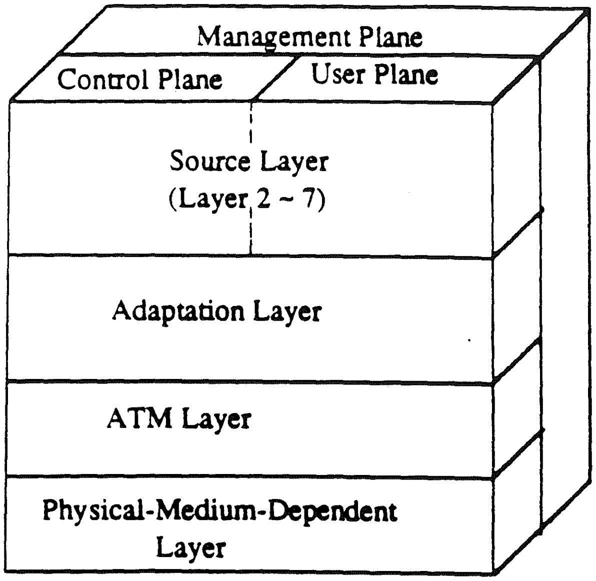


Figure 2.16: B-ISDN ATM Protocol Model

protocol architecture may be too heavy, i.e., it involves too much processing, and thus, a new protocol architecture is needed.

Layered architectures for ATM networks have been studied in [HM89, EGL88, RIDE88, VV88, FRAN87], and some agreement has been made in the CCITT Study Group XVIII meeting held in June 1989. Figure 2.16 [CCITT89c] depicts a B-ISDN ATM protocol model. In this figure, the protocol hierarchy consists of the physical-medium-dependent (PMD) layer, the ATM layer, the adaptation layer, and the higher service layer. Note that functional layering in the B-ISDN protocol model does not follow the OSI model [CCITT90].

The physical-medium-dependent (PMD) layer[4] underlies the ATM layer. As mentioned above, CCITT will standardize two physical interfaces to B-ISDN, one

[4] Also referred to as transmission layer.

based on SONET and the other based on a variation of ATD. This layer is responsible for the proper bit transmission and performs functions which are necessary to insert/extract the cell flow into/out-of a transmission frame. This layer is also responsible for electro-optical conversion since in B-ISDN, the physical medium is optical fiber.

The ATM layer contains all the details of the ATM technique, and it is common to all services. This layer is physical medium independent, and thus, it is independent of the underlying PMD layer. The data unit of this layer is an ATM cell, and the ATM layer performs the cell header functions. As discussed in subsection 2.4.1, major functions of the header include cell routing based on VCI/VPI and error detection on the header based on HEC. This layer also performs cell-based multiplexing/demultiplexing and cell delineation. The information field of an ATM cell is passed transparently through the ATM layer, and no processing, including error control, is performed on the information field at the ATM layer.

The adaptation layer and the higher layers of the ATM protocol model are service-dependent. The boundary between the ATM layer and the adaptation layer corresponds to the differences between functions applied to the cell header and functions applied to the information field [HM89]. The adaptation layer provides the higher service layer with the necessary functions which are not provided by the ATM layer, for instance, preserving timing, data frame boundaries and source clock. Functions of the adaptation layer are further described in the following.

Four service classes are defined at the adaptation layer. See Table 2.2 [VICK90]. Class 1 services correspond to constant bit rate (CBR) services. Constant bit rate audio and video belong to this class. Class 2 services corresponds to variable bit rate (VBR), connection-oriented services. Examples of class 2 services are variable

	Class 1	Class 2	Class 3	Class 4
Timing between source and destination	related		not related	
Bit rate	constant	variable		
Connection mode	connection-oriented			connectionless

Table 2.2: Service Classes

bit rate audio and video. For class 1 and 2 services, timing between source and destination needs to be related. Class 3 services also correspond to VBR connection-oriented services, but the timing between source and destination need not be related. Connection-oriented data and signaling data are examples of class 3 services. Class 4 services correspond to VBR connectionless services, and connectionless data belongs to this class.

The adaptation layer is divided into two sublayers, the Segmentation and Reassembly (SAR) sublayer and the Convergence (CS) sublayer; these two sublayers provide different functions for each of the four service classes. The following description gives a possible implementation of the adaptation layer for class 3 and 4 services. Two modes of adaptation service are provided for class 3 and 4 services: Message mode and Streaming mode. Message mode service is used for framed data, whereas Streaming mode service is used for low speed continuous data with low delay requirements. For Message mode service (see Figure 2.17 [VICK90]), the Convergence sublayer accepts a Service Data Unit (SDU) from the higher service layer. A SDU is a service-specific, higher layer information unit. It then prepends a 4-octet header (CS_PDU Header) to the SDU, pads the SDU (0 to 3-octet PAD) to make it an integral multiple of 32-bits, and appends a 4-octet trailer (CS_PDU Trailer). The

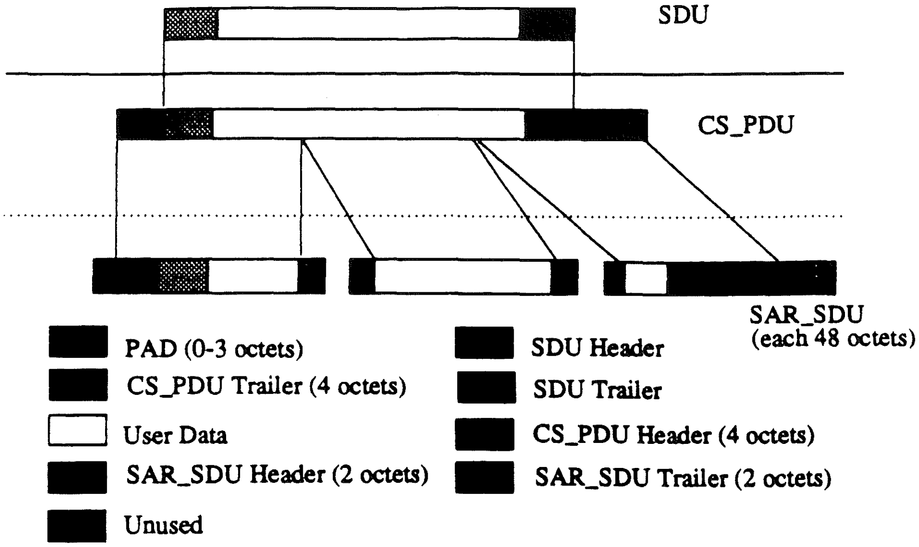


Figure 2.17: Message Mode Service

functions of the header and trailer fields include service indication and cell loss detection. The CS_PDU Header, SDU, PAD, and CS_PDU Trailer structure is referred to as a CS_PDU (Convergence Sublayer Protocol Data Units). After the trailer is appended, the CS_PDU is passed to the next sublayer, SAR sublayer, for segmentation. The SAR sublayer accepts a CS_PDU from the Convergence sublayer and segments it into N 44-octet SAR sublayer Service Data Unit (SAR_SDU) Payloads; thus, the last SAR_SDU Payload may have some unused portion. It then prepends a 2-octet header (SAR_SDU Header) to the SAR_SDU Payload and appends a 2-octet trailer (SAR_SDU Trailer) to the SAR_SDU Payload. The functions of the header and trailer fields include:

- Segmentation/reassembly
- Identification of segment type (e.g., a beginning of a message (BOM), a continuation of a message (COM), an end of a message (EOM), or a single SAR_SDU message (SSM))
- Identification of a message

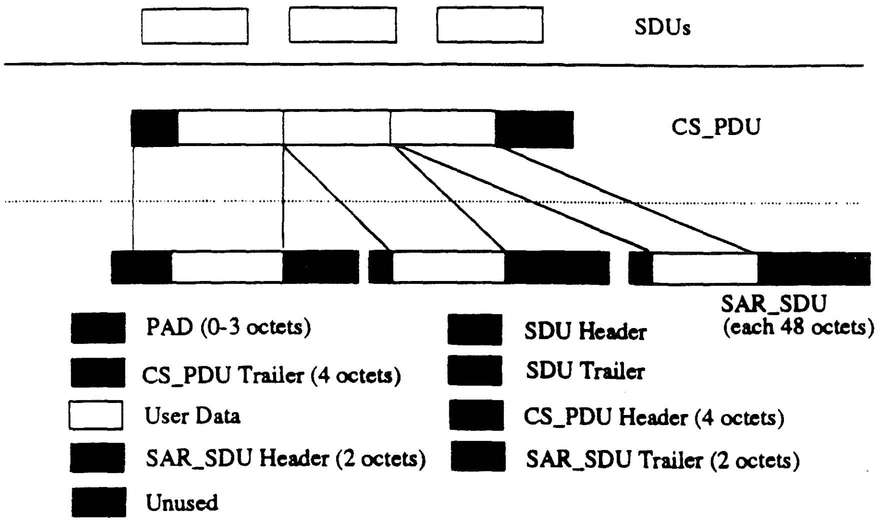


Figure 2.18: Streaming Mode Service

- Indication of partially filled segment
- Bit error detection for the entire contents of the SAR_SDU

The SAR_SDU Header, SAR_SDU Payload, and SAR_SDU Trailer structure is referred to as a SAR_SDU. After the trailer is appended, the SAR_SDU is passed to the ATM layer.

For Streaming mode service (see Figure 2.18 [VICK90]), unlike in Message mode service where one CS_PDU consists of one SDU, a CS_PDU may consist of several SDUs. As in Message mode service, a CS_PDU Header (4 octets), a PAD (0-3 octets) and a CS_PDU Trailer(4 octets) are added to complete a CS_PDU. Then, the SAR sublayer segment the CS_PDU into N 44-octet SAR_SDU Payloads such that each SDU is contained in a separate SAR_SDU. Therefore, unlike in Message mode service where only the last segment may contain some unused portion, in Streaming mode service any segment can have some unused portion. As in Message mode service, the SAR sublayer adds a 2-octet header and a 2-octet trailer to a SAR_SDU Payload. The functions of each field are identical to those of the Message mode service.

The possible CBR service (Class 1 service) adaptation functions include:

- Source clock frequency recovery
- Forward error correction
- Time stamping
- Sequence number processing
- Handling of lost or misdelivered cells

Adaptation layer functions for Class 2 services are not well defined. For further discussion, refer to [VICK90].

The higher service layer provides separate functions for the User plane and the Control plane, whereas lower layers (i.e., PDM layer, ATM layer and adaptation layer) provide functions common for those two planes. The Control plane is responsible for signaling, whereas the User plane is responsible for the transfer of user information. In the Control plane, much of the structure of existing N-ISDN (Narrowband ISDN) is maintained with some future enhancements. For example, an error detection/correction and flow control protocol in case of overload can be derived from the ISDN LAP-D protocol, and the call control (call establishment/release) mechanism is compatible to the N-ISDN protocol I.451. However, for the User plane, most of the link-by-link layer protocols are removed or pushed to higher edge-to-edge layers. For example, no link-by-link error control is provided.

2.5 Chapter Summary

Among the techniques proposed for B-ISDN transfer mode, ATM is considered to be the most promising transfer technique because of its flexibility and efficiency. In this chapter, a number of topics related to ATM networks were surveyed and reviewed. The topics covered are modeling of various traffic sources, congestion control, priority schemes to support multiple classes of traffic and standardization of ATM. The following conclusions were made:

- The cell arrival process for data sources can be modeled by a simple Poisson process. However, voice or video sources require more complex processes because of the correlation among cell arrivals.
- Due to the effects of high-speed channels, preventive control is more effective in ATM networks than reactive control.
- In order to satisfy diverse service and performance requirements of multimedia traffic, separate control mechanisms should be used for different traffic classes. Priority schemes effectively support multiple classes of traffic by providing different control mechanism to different priority services.
- In CCITT Recommendation I.121, ATM is accepted as the transfer mode for B-ISDN. Although some consensus has been made on some aspects of ATM, a number of issues related to ATM still need further study for standardization.

Chapter 3

Protocol-Processing Overhead in Error Recovery Schemes

In this chapter, we investigate one of the key research issues explained in Chapter 1: what is the cost of protocol-processing overheads on network performance? As described in the previous chapters, due to the increased overhead of protocol processing in high-speed networks, it becomes essential to re-evaluate the currently used control schemes. In this chapter, we investigate the effects of protocol-processing overhead on the performance of error recovery schemes. Protocol-processing time, such as time required to detect and correct errors via retransmissions, is comparatively large in high-speed networks and becomes dominant in determining the packet transfer delay across a networks. We focus on the widely accepted error recovery scheme in high-speed networks – namely, the scheme where retransmissions of erred packets only take place between source and destination nodes (edge-to-edge error recovery scheme). We obtain an approximation for the Laplace transform for the distribution of the end-to-end packet transfer delay, considering processing time required for error recovery. We also evaluate the performance of an alternative error recovery scheme; namely, a scheme where retransmissions take place between adjacent nodes (link-by-link error recovery scheme) and compare the performance of an edge-to-edge scheme with this alternative.

There exists some previous work [BKTV88, LAM76, IP80, KUHL83] on error recovery schemes. Despite the importance of the protocol-processing overhead, none of these previous work addressed this issue. In [BKTV88], link-by-link and edge-to-edge error control schemes were examined in the setting of a high-speed network. The effects of propagation delays were considered, but the protocol-processing overhead was not considered. In [LAM76, IP80, KUHL83], error control schemes were examined in the context of low-speed data networks. In [LAM76], a model was developed only for a link-by-link scheme, and the effects of propagation delays were considered in this model. The effect of protocol-processing overhead was not considered. In [IP80], link-by-link and edge-to-edge schemes were investigated, but the effects of propagation delays and protocol-processing times were not considered. In [KUHL83], link-by-link and edge-to-edge schemes were compared, but the assumptions made in the model inherently favored the edge-to-edge scheme; in the edge-to-edge scheme, each message was segmented whereas in the link-by-link scheme, each message was transmitted as a unit.

Our work described in this chapter differs from the above studies in that our work considers the protocol-processing overhead. Furthermore, unlike the previous work [BKTV88, LAM76, IP80, KUHL83], we use a layering architecture for high-speed networks and take a vertical view of the layers to investigate error recovery schemes at various layers of a protocol. Further, the Go-Back-N and Selective-Repeat procedures are considered in this work. All the previous work [BKTV88, LAM76, IP80, KUHL83] considers only the Selective-Repeat retransmission scheme.

Very little previous work [BRAD88, BS90a, NL90] considered the protocol-processing overhead. In [BRAD88], link-by-link and edge-to-edge schemes were studied through simulations for existing X.25 packet networks. In [BS90a], link-by-link and edge-to-edge schemes were compared, through simulations, in the setting of a high-speed network. No analysis was presented in [BRAD88, BS90a]. In [NL90], analytic results that considered processing times were presented for link-by-link and edge-to-edge schemes. However, only the Selective-Repeat retransmission scheme was considered, and the layering architecture was not taken into account in [NL90].

The rest of the chapter is organized as follows: in Section 3.1, we describe the network architecture to be investigated. In Section 3.2, we develop a queueing network model to evaluate error recovery schemes. In Section 3.3, an approximate analysis of the model developed in Section 3.2 is presented, and the end-to-end packet transfer delay is obtained. In Section 3.4, numerical results are presented to show the performance trade-offs between the error recovery schemes. In Section 3.5, a summary of this chapter is given. Finally, in Appendix A, some of the assumptions made in the analytic model are discussed.

3.1 Network Architecture To Be Investigated

To investigate the effects of protocol-processing time on the network performance, we assume the following hypothetical layering architecture for high-speed networks [HOBE83], and compare it with conventional architecture. We look only at the lower three layers of the OSI reference model.

In our hypothetical layering architecture, the basic transport mechanism is provided by the lower three levels of the protocol (1-, 2-, and 3-lower) applied to transmissions across each network link. The packet transport sublayer (level 3-upper) provides edge-to-edge communication within the network between source and destination nodes. The higher level end-to-end functions discussed in the OSI model (i.e., the transport layer and above) appear as higher layers above this basic transport mechanism and are not discussed here. The function of each layer in our layering architecture is described below.

- Level 1 - Physical Layer (P-layer) specifies the electrical characteristics and representation of transmitted bits.
- Level 2 - Link Layer (L-layer) performs several functions necessary for successful transmission between network nodes. These functions include frame delimiting and bit pattern transparency. As a major departure from conventional architecture (e.g., HDLC), error recovery procedures are not included in this level.
- Level 3 - This layer consists of the following two sublayers.
  - Level 3 Lower - Packet Network Sublayer (PN-layer) is the lower sub-layer of layer 3. The primary function provided here is the routing of packets.
  - Level 3 Upper - Packet Transport Sublayer (PT-layer) is the upper sublayer of layer 3 and performs edge-to-edge error recovery.

In order to reduce the protocol-processing overhead, error recovery between adjacent switching nodes is not performed in the above layering architecture. Instead, reliable data communication through the network is provided at the edge of the network (i.e., at level 3 upper); any detected errors are corrected with a peer communication between the source and destination nodes.

As mentioned in the previous chapters, propagation delay, as well as the time required for protocol-processing, is an important factor that decides network performance; propagation delay, which is assumed to be negligible in existing networks, becomes another dominant factor in determining the packet transfer delay across a network. In our analysis, both processing time and the propagation delay are considered in obtaining the end-to-end packet transfer delay.

3.2 Analytic Model

3.2.1 Protocol Layer Model

We seek to evaluate protocol-processing overhead in high-speed network environments, and thus, each layer at a switching node, rather than the switching node as a whole, is modeled as a queueing system in our analytic model [MT87]. In our model, the time a packet spends at a layer represents the overhead at that layer due to processing of the protocol itself and the operating system that supports the protocol's execution (e.g., process scheduling, data copying, buffer management, and timer management). Throughout Chapter 3, we refer to the time that a packet spends at a layer due to this protocol/OS combined overhead as the protocol processing time at that layer.

Table 3.1 focuses on the overhead due to processing of the protocol itself and lists the protocol functions at each layer for edge-to-edge and link-by-link error recovery schemes. Note that the operating system's overhead is not included in this table. In an edge-to-edge scheme, L-layer performs frame delimitation and transmission, but hop-by-hop error recovery is not performed. PN-layer performs routing function. PT-layer performs end-to-end error recovery. In a link-by-link scheme, L-layer performs frame delimitation and transmission, and corrects errors on a hop-by-hop basis. PN-layer routes packets, and PT-layer performs no specific function.

In addition to the protocol overhead, there exists the overhead due to the operating system and its associated protocol execution environment at each layer. (Note that this overhead is not listed in Table 3.1.) This overhead includes, among others, time required to read the local state information (of a connection) from a state table stored in memory, the time required to make a copy of buffer contents, and the sleep

	Layer	*Functions*	*Error Recovery Schemes*	
			E-to-E	L-by-L
Layer 3	PT-layer	Edge-to-Edge Packet Error Recovery	yes	no
	PN-layer	Routing	yes	yes
Layer 2	L-layer	Link-by-Link Frame Error Recovery	no	yes
		Frame Delimitation Frame Transmission	yes	yes

E-to-E:	Edge-to-Edge Error Recovery Scheme
L-by-L:	Link-by-Link Error Recovery Scheme

Table 3.1: Protocol Model

or idle time of a communication process due to an interrupt from a higher priority process. This operating system overhead, as well as the overhead due to processing of the protocol itself, will be considered in our analysis. (More detailed discussion on the protocol and the operating system overhead is found in the Appendix A.)

We assume that data units at L-layer and at PN- and PT-layers are of the same length, and thus, in our analysis, we do not distinguish frames from packets. It is also worth noting that in the analysis, variables with subscript 1, 2, and 3 are associated with L-layer, PN-layer, and PT-layer respectively.

3.2.2 Queueing Network Model

Figure 3.1 shows the queueing network model for link-by-link and edge-to-edge error recovery schemes based on the protocol architecture described in the previous section. In both link-by-link and edge-to-edge schemes, higher layers pass new packets to PT-layer at a source station. We assume that new packets arrive at PT-layer according to a Poisson process with rate λ_3. (In this work, we focus primarily on data applications and represent new packet arrivals as a Poisson process. Detailed discussion on the validity of this assumption is given in the Appendix A.) We assume that the packet length is, on the average, P bits, and the average transmission time of a packet is P/V sec, where V is the speed of the physical channel.

We further assume that the protocol-processing time at each layer follows an exponential distribution and that the protocol-processing times at different layers are independent. Recall that the protocol-processing time refers to the overhead due to processing of the protocol itself and the operating system that supports the protocol's execution. As mentioned in subsection 3.2.1, various factors affect this protocol/OS combined overhead. It is reasonable to assume that these overhead factors collectively provide random amounts of overhead, and thus, we assume that the protocol-processing time follows an exponential distribution (with the corresponding rate defined in the following paragraph). (The exponential protocol-processing time is further justified in the Appendix A.) Furthermore, since the functions that different layers perform, as well as the operating system overheads, are independent between different layers, each time a packet joins a queue in our queueing network model, its protocol-processing time is determined afresh from a corresponding exponential distribution.

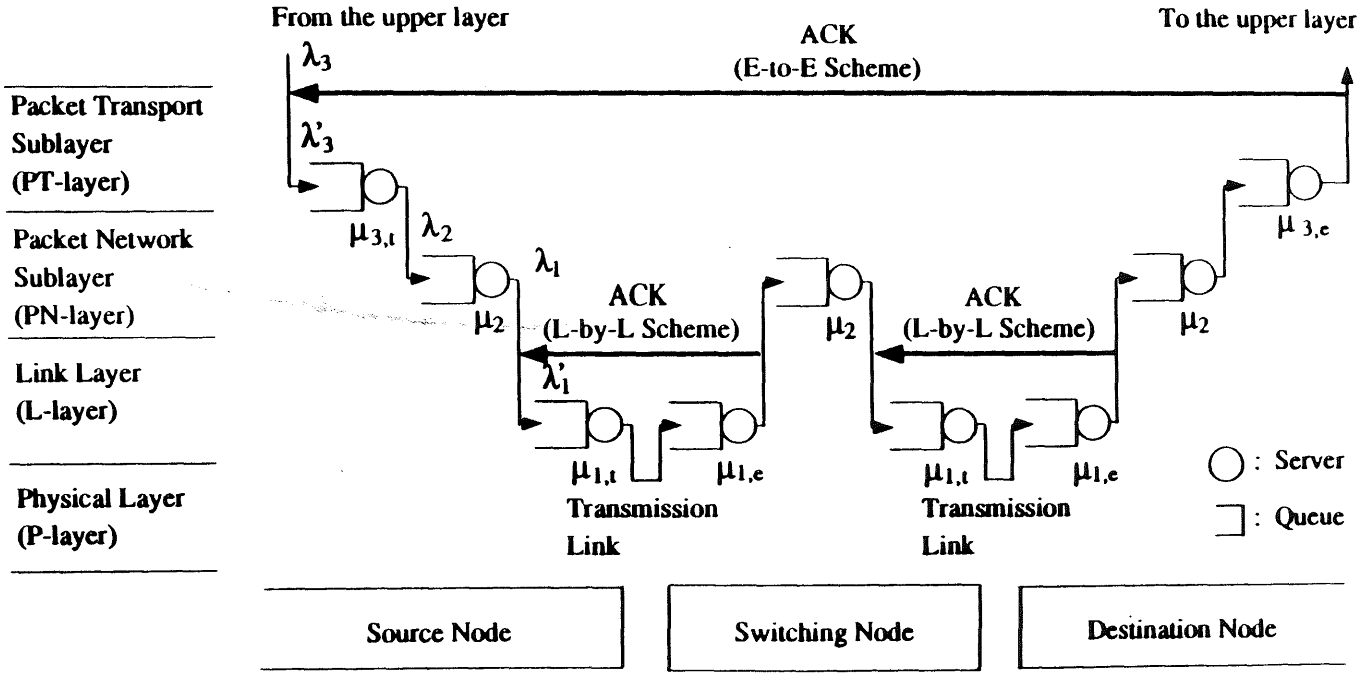


Figure 3.1: Queueing Network Model for Link-by-Link and Edge-to-Edge Error Recovery Schemes

Packets are stored and processed at PT-layer and then passed to PN-layer. The packet processing rate at PT-layer of a transmitting node is assumed to be $\mu_{3,t}$. (The subscript t stands for "transmitting node processing rate.") PN-layer makes routing decisions and passes packets to L-layer. The processing rate at PN-layer is assumed to be μ_2. Note that since the same rate μ_2 is assumed at PN-layer throughout the network, the subscript t is not necessary. At L-layer, packets are stored, processed and then transmitted through a physical transmission link. The processing rate at L-layer is assumed to be $\mu_{1,t}$.

In a link-by-link scheme, L-layer stores incoming packets at an intermediate switching node and examines the packets for errors. If no error is detected, the receiver immediately sends an ACK back to the sender. This is indicated by a feedback line (an arrow) between two adjacent L-layers in Figure 3.1. In case of error, no ACK is sent to the sender, and the sender retransmits the packet after the specified link-by-link time-out period. The processing rate at L-layer for error detection and ACK creation is assumed to be $\mu_{1,e}$. (The subscript e stands for processing rate for "error detection and ACK creation.") The procedure for retransmitting erred packets is described in the next subsection. If no error is found at L-layer, the packet is passed to PN-layer, where a routing decision is made. If the packet is addressed to some other node, it is passed down to L-layer for further transmission to the next node on the path to the destination. If the packet is destined for that node, it is passed up to PT-layer, and PT-layer immediately forwards the packet to the higher layers. Note that in a link-by-link scheme, no processing is done at PT-layer at destination.

In an edge-to-edge scheme, when an intermediate L-layer queue receives a packet, it immediately passes the packet to PN-layer for routing. L-layer does not perform any error checking or ACK transmission on incoming packets (i.e., $\mu_{1,e} = \infty$). As in a link-by-link scheme, if the packet is addressed to some other node, PN-layer passes it down to L-layer for further transmission to the next node. If the packet is destined for that node, PN-layer passes it up to PT-layer. In an edge-to-edge scheme, PT-layer at the destination node performs error checking. If no error is detected, the destination node immediately sends an ACK back to the source node. This is indicated by a feedback line from a destination to a source node in Figure 3.1. If there is an error in a packet, the source node retransmits the packet after the specified edge-to-edge time-out period. The rate of the processing to detect errors and create ACKs at PT-layer is assumed to be $\mu_{3,e}$.

In the model, propagation delay along the links is also considered. For simplicity, constant propagation delay D_{prop} is assumed between adjacent nodes (i.e., internodal distance is constant). Note that there may be some interfering traffic at each switching node. The effect of this interfering traffic can easily be modeled by reducing the communication capacity (i.e., a service rate of each queueing system in our model) as in [BKTV88, PS75].

3.2.3 Errors, Retransmissions, and Time-outs

In a link-by-link scheme, in case of error, a packet may be transmitted up to M_1 times (initial transmission and up to $M_1 - 1$ retransmissions) at L-layer between two adjacent nodes. In an edge-to-edge scheme, a packet may be transmitted up to M_3 times (initial transmission and up to $M_3 - 1$ retransmissions) at PT-layer between a source and a destination node. Note that in our analysis, M_1 and M_3 could be either finite or infinite. If M_1 is finite, a packet received at L-layer still has an error (or errors) with probability $p_1^{M_1}$ at the end of M_1 transmissions, where p_1 is the probability that a packet suffers an error (or errors) on a link. [1] We assume packets that still have errors after M_1 transmissions are discarded and will not be forwarded to the next node. Therefore, in the link-by-link scheme, packet loss probability ε across a network becomes

$$\varepsilon = 1 - (1 - p_1^{M_1})^l \tag{3.1}$$

where l is the number of hops between source and destination nodes. (If M_1 is infinite, this loss probability approaches zero.)

Similar to the link-by-link case, if M_3 is finite in an edge-to-edge scheme, a packet received at PT-layer of the destination node still has an error (or errors) with probability $P_3^{M_3}$ at the end of M_3 transmissions, where p_3 is the probability that a packet arrives at PT-layer of the destination node with an error (or errors). p_3 is given by $1 - (1 - p_1)^l$, where $(1 - p_1)^l$ is the probability that no error occurs in a packet in l number of hops. Assuming the packets that still have errors after M_3 transmissions are discarded at destination PT-layer, packet loss probability ε across a network in the edge-to-edge scheme becomes

$$\varepsilon = p_3^{M_3} = (1 - (1 - p_1)^l)^{M_3}. \tag{3.2}$$

(If M_3 is infinite, this error probability approaches zero.)

For retransmission of erred packets at L-layer (in a link-by-link scheme) and at PT-layer (in an edge-to-edge scheme), we consider both Go-Back-N and Selective-Repeat procedures in the analysis.

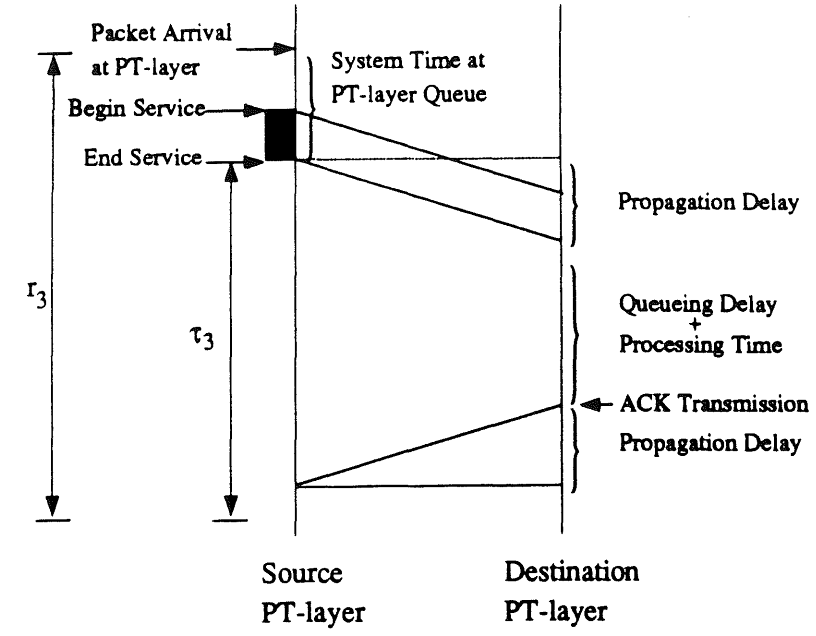
In the Go-Back-N procedure, a sender can send packets up to a given window size (N) without receiving acknowledgements from a receiver. When the receiver

[1] If b is the bit error rate on a link and if x is the number of bits in a packet, then the probability p_1 that a packet is received in error is $1 - (1 - b)^x$ under the assumption of *iid* errors. For large x, this quantity is approximately $1 - e^{-bx}$.

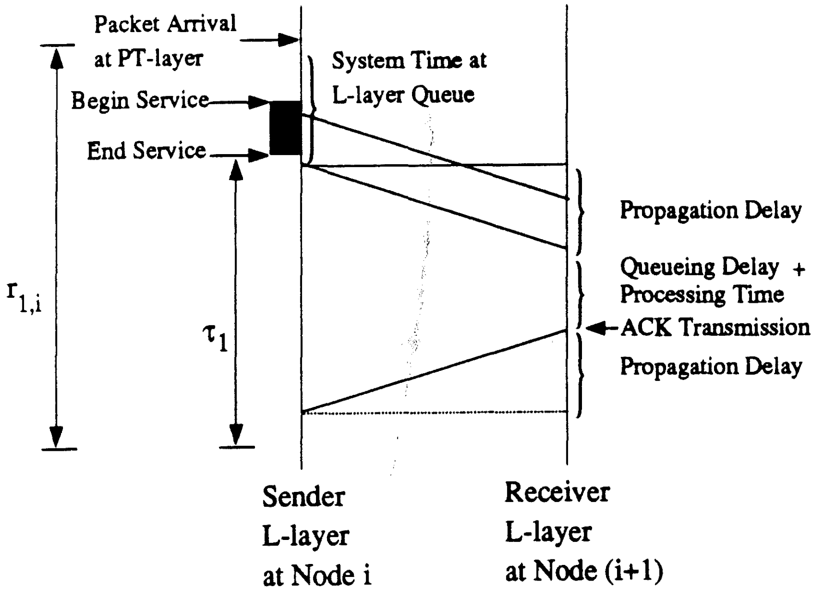
receives a packet with no errors, it immediately sends an ACK back to the sender. If an ACK is not received within a specified time-out period, the packet is assumed lost or erred, and the sender retransmits all the packets starting with the lost/erred packet. In the Selective-Repeat procedure, when the time-out period expires, the sender retransmits only the lost/erred packet.

Constant time-out periods τ_3 and τ_1 are used for retransmissions in the edge-to-edge and link-by-link schemes, respectively (see Figure 3.2). The edge-to-edge time-out period begins immediately after PT-layer passes a packet to PN-layer. The link-by-link time-out period begins immediately after transmission of a packet is completed at the sender L-layer. In both the edge-to-edge and link-by-link schemes, it is assumed that a time-out occurs only when a packet is, indeed, lost or erred. In reality, for a given time-out period, long queueing delays within a network can cause time-outs even when packet transmission is successful. For simplicity, this case is not considered in the analysis (The same assumption is made in [BKTV88, LAM76, IP80, KUHL83, NL90].)

We further define r_3 as the average time interval from an arrival of a packet at the source PT-layer queue to the end of the edge-to-edge time-out period, and r_1 as the average time interval from an arrival of a packet at a sender L-layer queue to the end of the link-by-link time-out period (see Figure 3.2). From Figure 3.2 (a), it is easy to see that r_3 consists of the average system time (i.e., the average queueing delay plus the service time) at the source PT-layer queue and the edge-to-edge time-out period τ_3. From Figure 3.2 (b), r_1 consists of the average system time (i.e., the average queueing delay plus packet transmission time) at a sender L-layer queue and the link-by-link time-out period τ_1. Note that the average queueing delay at a sender L-layer queue differs at each node since the arrival rate to each sender L-layer queue changes due to the packet discarding. (Packets that still have errors after M_1



(a) Time-out Period in the Edge-to-Edge Scheme


Packet Arrival at PT-layer
Begin Service
End Service
System Time at L-layer Queue
Propagation Delay
Queueing Delay + Processing Time
ACK Transmission
Propagation Delay
$r_{1,i}$
τ_1
Sender L-layer at Node i
Receiver L-layer at Node (i+1)

(b) Time-out Period in the Link-by-Link Scheme

Figure 3.2: Time-outs

transmissions are discarded.) Therefore, each node has different values of r_1. We let $r_{1,i}$ denote r_1 at node i.

3.3 Analysis

3.3.1 Assumptions

In subsection 3.2.2, we assumed that each time a packet joins a queue in our queueing network model, its protocol-processing time is determined afresh from a corresponding exponential distribution. As explained earlier, this is due to the fact that the functions different layers perform as well as the operating system overhead are independent between layers.

In order to make the analysis tractable, we also employ the following assumption in our analysis:

- In an edge-to-edge scheme, the aggregated arrivals of new and retransmitted packets at a PT-layer source queue are assumed to follow a Poisson process. In a link-by-link scheme, it is assumed that at each sender L-layer queue the aggregated arrivals of packets passed from PN-layer and retransmitted packets follow a Poisson process.

The following explains the validity of the above assumption. We focus on high-speed networks that utilize very high-speed and high-quality channels (i.e., optical fibers). Optical fibers can provide very low error probability (i.e., 10^{-9} to 10^{-13} bit error rate), and thus, packet retransmissions due to errors are very scarce in such networks. Furthermore, as we will see in the numerical example section (Section

3.4, Figure 3.5), in case of errors, almost all the errors are corrected by at most two transmissions (initial transmission and one retransmission), and thus, packet discarding will rarely happen. Occasional retransmissions and losses will not severely destroy the Poisson property of an input process. The accuracy of this assumption is verified through extensive simulations in subsection 3.4.2.

In the analysis, we also assume that retransmissions do not have priority over transmissions of new packets and that errors do not occur in ACK packets. Furthermore, it is assumed that each node has an infinite buffer. In the case where Go-Back-N is used, an infinite window size is assumed.

With the assumptions explained in this section, our queueing network model becomes a Jackson type network [JACK63], and each queue in the network behaves as if it were an independent M/M/1 system. In subsections 3.3.2 and 3.3.3, a link-by-link scheme and an edge-to-edge scheme are analyzed, respectively.

3.3.2 Link-by-Link Scheme

In this subsection, we focus on a link-by-link error recovery scheme and analyze its performance.

Effective Bit Rate

We first consider a queueing system at L-layer of a sending node (i.e., a queue with the service rate $\mu_{1,t}$ in Figure 3.3.) Note that when the Go-Back-N retransmission procedure is employed, the sending node retransmits all the packets in its L-layer queue starting with the lost/erred packet when the time-out period τ_1 expires. With this in mind, if the initial transmission of a packet fails at node i, at the first retransmission (i.e., the second transmission) of the erred packet, the packets that arrived

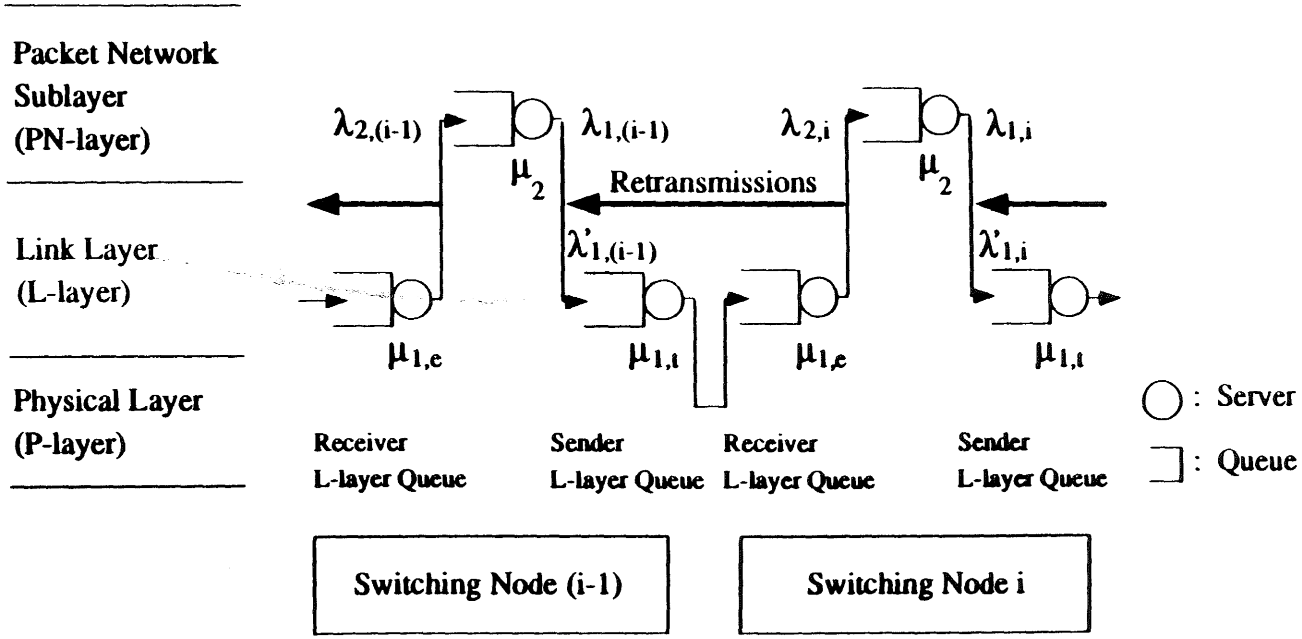


Figure 3.3: Arrival Rates at PN-layer and L-layer Queues

during $r_{1,i}$ plus the erred packet itself are retransmitted (see Figure 3.2 (b)). Thus, an average of $r_{1,i}\lambda_{1,i}+1$ packets are transmitted at the second transmission of the erred packet, where $\lambda_{1,i}$ (Figure 3.3) is the packet arrival rate at L-layer at node i. At the j-th transmission of an erred packet, the packets that arrived during $(j-1)r_{1,i}$ are transmitted along with the erred packet. Letting $N_{1,i}$ denote $r_{1,i}\lambda_{1,i}$, an average of $(j-1)N_{1,i}+1$ packets are transmitted at node i at the j-th transmission of an erred packet when Go-Back-N is employed. On the other hand, when Selective-Repeat is employed, only the erred packet is retransmitted, and thus, it is easy to see that $N_{1,i}=0$ for all i's.

Let $P(k)$ be the probability that a packet requires k transmissions (an initial transmission and $(k-1)$ retransmissions) before it is either passed up to PN-layer or discarded at the receiving L-layer (due to the limitation on the maximum number of retransmissions allowed), where $1 \le k \le M_1$. It is easily shown that $P(k)$ is given by

$$P(k) = \begin{cases} q_1 p_1^{k-1} & (1 \le k < M_1) \\ p_1^{M_1-1} & (k = M_1) \end{cases} \tag{3.3}$$

where $q_1 = 1 - p_1$. If a packet requires k transmissions (including the initial transmission), an average of $\sum_{j=1}^{k}\{(j-1)N_{1,i}+1\}$ packets are transmitted until the packet is accepted at the receiving L-layer. Therefore, the average number of packets $Y_{1,i}$ to be transmitted until a packet is finally accepted at receiving L-layer becomes

$$\begin{aligned} Y_{1,i} &= \sum_{k=1}^{M_1}[\sum_{j=1}^{k}\{(j-1)N_{1,i}+1\}]P(k) \\ &= N_{1,i}\frac{p_1\{1-M_1p_1^{M_1-1}+(M_1-1)p_1^{M_1}\}}{(1-p_1)^2}+\frac{1-p_1^{M_1}}{1-p_1} \end{aligned} \tag{3.4}$$

From this we obtain the effective packet arrival rate $\lambda'_{1,i}$ (i.e., the aggregated arrival rate of the new packets from PN-layer and the retransmitted packets) at the sender

L-layer queue at node i (Figure 3.3):

$$\lambda'_{1,i} = \lambda_{1,i} Y_{1,i} = \lambda_{1,i}[N_{1,i} \frac{p_1\{1 - M_1 p_1^{M_1-1} + (M_1 - 1)p_1^{M_1}\}}{(1-p_1)^2} + \frac{1 - p_1^{M_1}}{1 - p_1}]. \tag{3.5}$$

We next consider the source node (see Figure 3.1). At the source node, we assume that packets arrive at PT-layer with the rate λ_3. Since the output from the PT-layer queue is the input to its PN-layer queue, the packet arrival rate λ_2 at the source PN-layer queue is $\lambda_2 = \lambda_3$. (Note that a link-by-link scheme is assumed in this section. Thus, there is no end-to-end retransmission.) Also, since the output from the PN-layer queue is the input to its source L-layer queue, the packet arrival rate λ_1 at the sender L-layer queue from the PN-layer queue becomes $\lambda_1 = \lambda_2 (= \lambda_3)$. The packets from PN-layer (at the rate of λ_1) and retransmissions of erred packets (indicated by the feedback line in Figure 3.1) collectively form the arrival to the L-layer queue at the source node. The rate of this aggregated packet arrivals, λ'_1, is given by Eq.(3.5). Note that $\lambda_1 = \lambda_{1,1}$, $\lambda'_1 = \lambda'_{1,1}$, and $\lambda_2 = \lambda_{2,1}$.

At the intermediate nodes, since the packets with errors after M_1 transmissions are discarded at L-layer, the rate of packet arrivals at PN-layer from L-layer varies depending on the node (see Figure 3.3). Noting that $p_1^{M_1}$ is the probability that a packet is discarded (at node i) due to an error after M_1 transmissions, the packet arrival rate $\lambda_{2,i}$ at PN-layer at node i becomes

$$\lambda_{2,i} = \lambda_{1,(i-1)}(1 - p_1^{M_1}) \tag{3.6}$$

Further, the rate $\lambda_{1,i}$ at which packets are passed from PN-layer to sender L-layer at node i becomes $\lambda_{1,i} = \lambda_{2,i}$. Note that the rate of the aggregated arrivals of packets (packets from PN-layer at the rate of $\lambda_{1,i}$ and retransmissions due to errors) at sender L-layer is given by Eq.(3.5).

The effective utilization of each queue at node i becomes

- $\rho'_{1,i,t} = \lambda'_{1,i}/\mu_{1,t}$ at a sender L-layer queue at node i,
- $\rho'_{1,i,e} = \lambda'_{1,(i-1)}/\mu_{1,e}$ at a receiver L-layer queue at node i,
- $\rho_{2,i} = \lambda_{2,i}/\mu_2$ at a PN-layer queue at node i, and
- $\rho_{3,t} = \lambda_3/\mu_{3,t}$ at the PT-layer queue at the source node.

The utilization of the PT-layer queue $\rho_{3,e}$ at the destination is not considered since it is assumed that no processing is done at the destination PT-layer queue if the link-by-link scheme is used.

End-to-End Packet Transfer Delay

We define the end-to-end transfer delay of a packet as the time starting when a packet first enters the source PT-layer queue and ending when that packet leaves the destination PT-layer. This transfer delay consists of (1) the time spent at a source PT-layer queue, (2) the time spent at a source PN-layer queue, (3) the time spent at intermediate switching nodes, and (4) the time spent at the destination PN-layer queue. Note that the time spent at the destination PT-layer queue is not included since in a link-by-link scheme no processing is involved in PT-layer at the destination, and packets from PN-layer are immediately forwarded to the upper layer.

Let $B_3^*(s)$ be the Laplace transform for the distribution of the end-to-end packet transfer delay. $B_3^*(s)$ is obtained in the following way. Let $F_{3,t}^*(s)$ be the Laplace transform for the distribution of the time spent by a packet in the source PT-layer queue, and $F_{2,i}^*(s)$ be the Laplace transform for the distribution of the time spent by a packet in a PN-layer queue at node i. Further, let $\Phi_{1,i}^*(s)$ be the Laplace transform for the distribution of the time required for a packet to "hop" to the next node, i.e., the time starting when a packet first enters the sender L-layer queue at node i and ending when the packet is passed to the PN-layer queue at node $i+1$, including the

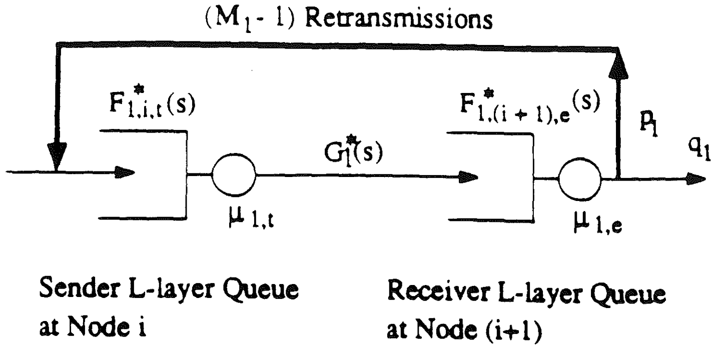


Figure 3.4: L-layer Queues

time incurred by retransmissions. Assuming that there are l hops from the source to the destination, $B_3^*(s)$ is given by

$$B_3^*(s) = F_{3,t}^*(s) F_{2,1}^*(s) \prod_{i=1}^{l} \{\Phi_{1,i}^*(s) F_{2,(i+1)}^*(s)\} \tag{3.7}$$

where $F_{3,t}^*(s)$ in the right hand side corresponds to the delay element (1) (see the definition of transfer delay given in the above paragraph); $F_{2,1}^*(s)$ corresponds to (2); and the last product term $\prod_{i=1}^{l}\{\Phi_{1,i}^*(s)F_{2,(i+1)}^*(s)\}$ corresponds to (3) and (4).

We first obtain $\Phi_{1,i}^*(s)$ in Eq.(3.7). Let us consider L-layer queues in two adjacent nodes: a sender L-layer queue at node i (with the service rate $\mu_{1,t}$) and a receiver L-layer queue at node $i+1$ (with the service rate $\mu_{1,e}$) (see Figure 3.4). Let $F_{1,i,t}^*(s)$ and $F_{1,(i+1),e}^*(s)$ be the Laplace transform for the distribution of the time that a packet spends at the sender L-layer queue and at the receiver L-layer queue, respectively. As discussed in subsection 3.3.1, in a link-by-link scheme it is assumed that at sender L-layer, the aggregated arrivals of packets passed from PN-layer and retransmitted packets follow a Poisson process. (This implies that the input to the receiver L-layer queue at the next node also follows a Poisson process, since the service at the sender L-layer queue is exponential.) Therefore, we have

$$F_{1,i,t}^*(s) = \frac{\mu_{1,t}(1-\rho'_{1,i,t})}{s+\mu_{1,t}(1-\rho'_{1,i,t})} \tag{3.8}$$

$$F^*_{1,(i+1),e}(s) = \frac{\mu_{1,e}(1 - \rho'_{1,(i+1),e})}{s + \mu_{1,e}(1 - \rho'_{1,(i+1),e})} \tag{3.9}$$

Here, we have used the Laplace transform for the system time distribution in an M/M/1 queue. (With an arrival rate λ, a service rate μ and $\rho = \lambda/\mu$, it is given by $\frac{\mu(1-\rho)}{s+\mu(1-\rho)}$ [SC81, WONG78].)

Assume a packet requires k transmissions to be accepted at receiver L-layer without errors ($1 \leq k \leq M_1$). This happens with probability $q_1 p_1^{k-1}$. In this case, during the first $k-1$ transmissions the packet goes through the sender L-layer queue $k-1$ times and the time-out $k-1$ times. At the last (successful) transmission, the packet goes through the sender L-layer queue, propagates along the link, and gets processed at the receiver L-layer queue. Since the time spent by a packet in the sender L-layer queue, a time-out period, the propagation delay on a link, and the time spend by a packet in the receiver L-layer queue are independent, the Laplace transform $\Phi^*_{1,i}(s)$ for the distribution of the sojourn time (time starting when a packet first enters the sender L-layer queue at node i and ending when the packet is passed to the PN-layer queue at node $i+1$, including the time incurred by retransmissions) becomes

$$\begin{aligned}
\Phi^*_{1,i}(s) &= \sum_{k=1}^{M_1}[q_1 p_1^{k-1}\{F^*_{1,i,t}(s)S^*_1(s)\}^{k-1}F^*_{1,i,t}(s)G^*_1(s)F^*_{1,(i+1),e}(s)]\frac{1}{\sum_{k=1}^{M_1} q_1 p_1^{k-1}} \\
&= \sum_{k=1}^{M_1}\{q_1 p_1^{k-1}S^*_1(s)^{k-1}F^*_{1,i,t}(s)^k G^*_1(s)F^*_{1,(i+1),e}(s)\}\frac{1}{1-p_1^{M_1}} \\
&= \frac{1}{1-p_1^{M_1}}G^*_1(s)F^*_{1,(i+1),e}(s)A^*_1(s)
\end{aligned} \tag{3.10}$$

where $A^*_1(s) = \sum_{k=1}^{M_1}\{q_1 p_1^{k-1}S^*_1(s)^{k-1}F^*_{1,i,t}(s)^k\}$; $S^*_1(s)$ is the Laplace transform for the link-by-link time-out period ($S^*_1(s) = e^{-s\tau_1}$); $G^*_1(s)$ is the Laplace transform for the propagation delay ($G^*_1(s) = e^{-sD_{prop}}$); and $\frac{1}{1-p_1^{M_1}}(= \frac{1}{\sum_{k=1}^{M_1} q_1 p_1^{k-1}})$ is a normalizing factor. Note that only delays for successful packets are considered. Note also that the time-out period and the propagation delay between two adjacent nodes are constant.

$F^*_{3,t}(s)$ and $F^*_{2,i}(s)$ in Eq.(3.7) are easily obtained in the following way. $F^*_{3,t}(s)$ is the Laplace transform of the distribution of the time spent by a packet in a source PT-layer queue, and $F^*_{2,i}(s)$ is the Laplace transform of the distribution of the time spent by a packet in a PN-layer queue at node i. From the same argument used to obtain $F^*_{1,i,t}(s)$, we have

$$F^*_{3,t}(s) = \frac{\mu_{3,t}(1-\rho_{3,t})}{s+\mu_{3,t}(1-\rho_{3,t})} \tag{3.11}$$

$$F^*_{2,i}(s) = \frac{\mu_2(1-\rho_{2,i})}{s+\mu_2(1-\rho_{2,i})} \tag{3.12}$$

By substituting Eqs.(3.10), (3.11) and (3.12) into Eq.(3.7), we can obtain $B^*_3(s)$, the Laplace transform for the distribution of the end-to-end packet transfer delay. From this Laplace transform, the average T, the second moment $T^{(2)}$, and the standard deviation T_σ of the end-to-end transfer delay of packets are obtained as follows:

$$T = -\frac{d}{ds}B^*_3(s)|_{s=0} \tag{3.13}$$

$$T^{(2)} = \frac{d^2}{ds^2}B^*_3(s)|_{s=0} \tag{3.14}$$

$$T_\sigma = \sqrt{T^{(2)} - T^2} \tag{3.15}$$

In the following, we define

$$\Theta' = \frac{d}{ds}\Theta^*(s)|_{s=0}, \qquad \Theta'' = \frac{d^2}{ds^2}\Theta^*(s)|_{s=0} \tag{3.16}$$

where $\Theta^*(s)$ is a Laplace transform of a distribution. From simple manipulation, T and $T^{(2)}$ become

$$T = -B'_3 = -F'_{3,t} - F'_{2,1} - \sum_{i=1}^{l}(\Phi'_{1,i} + F'_{2,(i+1)})) \tag{3.17}$$

$$\begin{aligned} T^{(2)} &= B''_3 \\ &= F''_{3,t} + 2F'_{3,t}F'_{2,1} + F''_{2,1} + 2(F'_{3,t} + F'_2)\sum_{i=1}^{l}(\Phi'_{1,i} + F'_{2,(i+1)}) \end{aligned}$$

$$+\sum_{i=1}^{l}\{\Phi''_{1,i} + F''_{2,(i+1)} - (\Phi'_{1,i})^2 - (F'_{2,(i+1)})^2\}$$
$$+\{\sum_{i=1}^{l}(\Phi'_{1,i} + F'_{2,(i+1)})\}^2 \tag{3.18}$$

where

$$\Phi'_{1,i} = G'_1 + F'_{1,(i+1),e} + \frac{A'_1}{1-p_1^{M_1}} \tag{3.19}$$

$$\Phi''_{1,i} = G''_1 + F''_{1,(i+1),e} + 2G'_1F'_{1,(i+1),e} + \frac{1}{1-p_1^{M_1}}\{A''_1 + 2A'_1(G'_1 + F'_{1,(i+1),e})\} \tag{3.20}$$

$$A'_1 = \frac{(M_1-1)p_1^{M_1+1} - M_1p_1^{M_1} + p_1}{q_1}S'_1 + \frac{1-(M_1+1)p_1^{M_1} + M_1p_1^{M_1+1}}{q_1}F'_{1,i,t} \tag{3.21}$$

$$\begin{aligned} A''_1 \ = \ & \frac{2M_1(M_1-2)p_1^{M_1-1} - M_1(M_1-1)p_1^{M_1-2} - (M_1-1)(M_1-2)p_1^{M_1} + 2}{q_1^2}(S'_1)^2p_1^2 \\ & + \frac{p_1\{-M_1(M_1+1)p_1^{M_1-1} + 2(M_1+1)(M_1-1)p_1^{M_1} - M_1(M_1-1)p_1^{M_1+1} + 2\}}{q_1^2} \\ & \cdot\{2S'_1F'_{1,i,t} + (F'_{1,i,t})^2\} + \frac{1+(M_1-1)p_1^{M_1} - M_1p_1^{M_1-1}}{q_1}S''_1p_1 \\ & + \frac{1-(M_1+1)p_1^{M_1} + M_1p_1^{M_1+1}}{q_1}F''_{1,i,t} \end{aligned} \tag{3.22}$$

$$F'_{1,i,t} = -\frac{1}{\mu_{1,t}(1-\rho'_{1,i,t})}, \qquad F'_{1,i,e} = -\frac{1}{\mu_{1,e}(1-\rho'_{1,i,e})} \tag{3.23}$$

$$F'_{2,i} = -\frac{1}{\mu_2(1-\rho_{2,i})}, \qquad F'_{3,t} = -\frac{1}{\mu_{3,t}(1-\rho_{3,t})} \tag{3.24}$$

$$F''_{1,i,t} = \frac{2}{\mu_{1,t}^2(1-\rho'_{1,i,t})^2}, \qquad F''_{1,i,e} = \frac{2}{\mu_{1,e}^2(1-\rho'_{1,i,e})^2} \tag{3.25}$$

$$F''_{2,i} = \frac{2}{\mu_2^2(1-\rho_{2,i})^2}, \qquad F''_{3,t} = \frac{2}{\mu_{3,t}^2(1-\rho_{3,t})^2} \tag{3.26}$$

$$S'_1 = -\tau_1, \qquad G'_1 = -D_{prop}, \qquad S''_1 = \tau_1^2, \qquad G''_1 = D_{prop}^2. \tag{3.27}$$

At this point, the only unknown factor in Eqs.(3.13), (3.14) and (3.15) is $r_{1,i}$. Since $r_{1,i}$ is the average time interval from a packet's arrival at the sender L-layer queue to the end of link-by-link time-out period, $r_{1,i} = -F'_{1,i,t} + \tau_1$. (Refer to the definition of $r_{1,i}$ in subsection 3.2.3 and also to Figure 3.2 (b).)

3.3.3 Edge-to-Edge Scheme

In this subsection, we focus on an edge-to-edge error recovery scheme and analyze its performance.

Effective Bit Rate

First, we obtain the effective packet arrival rate λ'_3 (i.e., the rate of the aggregated arrivals of the new packets and the retransmitted packets) at the PT-layer queue of the source node (see Figure 3.1). From the same argument used to obtain the effective packet arrival rate $\lambda'_{1,i}$ at the L-layer queue at node i in a link-by-link scheme (see Eq.(3.5)), we have

$$\lambda'_3 = \lambda_3[N_3 \frac{p_3\{1 - M_3 p_3^{M_3-1} + (M_3-1)p_3^{M_3}\}}{(1-p_3)^2} + \frac{1-p_3^{M_3}}{1-p_3}]. \tag{3.28}$$

where $N_3 = r_3\lambda_3$ for Go-Back-N, and $N_3 = 0$ for Selective-Repeat.

At the source node, since the output from the PT-layer queue is the input to its PN-layer queue and in turn the input to its L-layer queue, we have $\lambda'_3 = \lambda_2 = \lambda_1$. Note that with an edge-to-edge scheme, no packet will be dropped at the intermediate nodes, and thus, the packet arrival rate at intermediate switching nodes does not change along the path. Packets are dropped at the destination PT-layer only if they have errors after M_3 edge-to-edge retransmissions. Therefore, the effective utilization of each queue becomes

- $\rho_{1,t} = \lambda_1/\mu_{1,t}$ at a sender L-layer queue,
- $\rho_2 = \lambda_2/\mu_2$ at a sender PN-layer queue,
- $\rho'_{3,t} = \lambda'_3/\mu_{3,t}$ at the source PT-layer queue, and
- $\rho'_{3,e} = \lambda'_3/\mu_{3,e}$ at the destination PT-layer queue.

The utilization of a receiver L-layer queue $\rho_{1,e}$ at intermediate switching nodes is not considered since no processing is required at a receiver L-layer queue.

End-to-End Packet Transfer Delay

An argument similar to the one used in the link-by-link scheme applies to obtain the Laplace transform $B_3^*(s)$ for the distribution of the end-to-end packet transfer delay in the edge-to-edge scheme.

Assume a test packet requires k edge-to-edge transmissions to be accepted at the destination PT-layer without errors ($1 \leq k \leq M_3$). This happens with probability $q_3 p_3^{k-1}$ where $q_3 = 1 - p_3$. Recall that p_3 is the probability that a packet arrives at the destination PT-layer with an error (or errors).

During the first $k-1$ retransmissions, the test packet goes through the PT-layer queue at the source node $k-1$ times and an edge-to-edge time-out $k-1$ times. (Note that an edge-to-edge time-out starts right after the source PT-layer queue passes a packet to the PN-layer queue.) The Laplace transform for the distribution of the time spent for these $k-1$ retransmissions is $\{F_{3,t}^*(s)S_3^*(s)\}^{k-1}$. Here, $F_{3,t}^*(s)$ is the Laplace transform for the distribution of the time spent at the source PT-layer queue. Since the input to the PT-layer queue at the source node follows a Poisson process, $F_{3,t}^*(s)$ is easily obtained by using the corresponding ρ value in the Laplace transform for the system time distribution in an M/M/1 queue. $S_3^*(s)$ is the Laplace transform of the edge-to-edge time-out period, and is given by $S_3^*(s) = e^{-s\tau_3}$.

For the last (successful) transmission, the test packet experiences the following delays: (1) the time spent at the PT-layer queue at the source, (2) the time spent at the PN-layer queue at the source, (3) the time spent at intermediate switching nodes (4) the time spent at the PN-layer queue at the destination, and (5) the time spent at the PT-layer queue at the destination. The Laplace transform for the delay element (1) is $F^*_{3,t}(s)$. The Laplace transform $F^*_2(s)$ for the delay element (2) is easily obtained from the Laplace transform for the system time distribution in an M/M/1 queue since the input to the PN-layer queue at the source node follows a Poisson process. Assuming that there are l hops from the source to the destination, the Laplace transform for (3) and (4) combined is given by $\{F^*_{1,t}(s)G^*_1(s)F^*_2(s)\}^l$, where $F^*_{1,t}(s)$ is the Laplace transform for the time spent at a sender L-layer queue (and is obtained by using a corresponding ρ value in the Laplace transform for the system time distribution in an M/M/1 queue); $G^*_1(s)$ is the Laplace transform for the propagation delay ($G^*_1(s) = e^{-sD_{prop}}$); and $F^*_2(s)$ is the Laplace transform for the time spent at a PN-layer queue. Note that the time spent at the L-layer queue at a receiving node ($F^*_{1,e}(s)$) is not included since no processing is required at this layer in an edge-to-edge scheme. The Laplace transform $F^*_{3,e}(s)$ for the delay element (5) is obtained from the Laplace transform for the system time distribution in an M/M/1 queue, since the input to the PT-layer queue at the destination node follows a Poisson process.

From the above discussion, $B^*_3(s)$ is given by

$$\begin{aligned}
B^*_3(s) &= \sum_{k=1}^{M_3}[q_3p_3^{k-1}\{F^*_{3,t}(s)S^*_3(s)\}^{k-1}F^*_{3,t}(s)F^*_2(s)\{F^*_{1,t}(s)G^*_1(s)F^*_2(s)\}^lF^*_{3,e}(s)] \\
&\quad \cdot\frac{1}{\sum_{k=1}^{M_3} q_3p_3^{k-1}} \\
&= \sum_{k=1}^{M_3}\{q_3p_3^{k-1}S^*_3(s)^{k-1}F^*_{3,t}(s)^kF^*_2(s)^{l+1}F^*_{1,t}(s)^lG^*_1(s)^lF^*_{3,e}(s)\}\frac{1}{1-p_3^{M_3}} \\
&= \frac{1}{1-p_3^{M_3}}F^*_2(s)^{l+1}F^*_{1,t}(s)^lG^*_1(s)^lF^*_{3,e}(s)A^*_3(s) \qquad (3.29)
\end{aligned}$$

where $A^*_3(s) = \sum_{k=1}^{M_3}\{q_3p_3^{k-1}S^*_3(s)^{k-1}F^*_{3,t}(s)^k\}$; and $\frac{1}{1-p_3^{M_3}}(= \frac{1}{\sum_{k=1}^{M_3} q_3p_3^{k-1}})$ is a normalizing factor.

From Eq.(3.29), the average T and the second moment $T^{(2)}$ of the end-to-end packet transfer delay become

$$T = -B'_3 = -(l+1)F'_2 - lF'_{1,t} - lG'_1 - F'_{3,e} - \frac{A'_3}{1-p_3^{M_3}} \tag{3.30}$$

$$\begin{aligned} T^{(2)} &= B''_3 \\ &= (l+1)F''_2 + lF''_{1,t} + lG''_1 + F''_{3,e} + l(l+1)(F'_2)^2 + l(l-1)(F'_{1,t})^2 \\ &\quad + l(l-1)(G'_1)^2 + 2l(l+1)F'_2F'_{1,t} + 2l(l+1)F'_2G'_1 + 2(l+1)F'_2F'_{3,e} \\ &\quad + 2l^2F'_{1,t}G'_1 + 2lG'_1F'_{3,e} + 2lF'_{1,t}F'_{3,e} + \frac{2A'_3\{(l+1)F'_2 + lF'_{1,t} + lG'_1 + F'_{3,e}\}}{1-p_3^{M_3}} \\ &\quad + \frac{A''_3}{1-p_3^{M_3}} \end{aligned} \tag{3.31}$$

where $S'_3 = -\tau_3$ and $S''_3 = \tau_3^2$. The rest of the parameters in Eqs.(3.30) and (3.31) can be easily obtained using the methods used in subsection 3.3.2. r_3 is given by $-F'_{3,t} + \tau_3$ (see Figure 3.2 (a)).

3.4 Numerical Examples

In this section we show some numerical examples for the link-by-link and the edge-to-edge error recovery schemes. In subsection 3.4.1, the packet loss probability ε across a network is investigated. In subsection 3.4.2, the accuracy of the assumptions made in the analysis is verified through simulations. In subsection 3.4.3, the average end-to-end packet transfer delays for link-by-link and edge-to-edge schemes are presented, and the performance trade-offs are discussed to find an optimal error recovery scheme for high-speed networks.

Throughout this section, unless otherwise stated, we assume the following parameter values: the average packet length $P = 1000$ bits, the channel speed $V = 600$

Mbits/sec, and the average packet transmission time = $P/V \approx 1.666 \times 10^{-6}$ seconds. In the figures "E-to-E" represents edge-to-edge scheme, and "L-by-L" represents link-by-link scheme.

3.4.1 Packet Loss Probability

The packet loss probability ε across a network (i.e., the probability that a packet is discarded due to the limitation on the maximum number of retransmissions allowed) is a function of M_1 (the maximum number of packet transmissions allowed at L-layer) in a case where a link-by-link error recovery scheme is used, or a function of M_3 (the maximum number of packet transmissions allowed at PT-layer) in a case where an edge-to-edge scheme is used. Please refer to Eqs.(3.1) and (3.2).

Figure 3.5 shows the effect of the values of M_1 and M_3 on the packet loss probability across a network when 1000 bit-long packets are transferred through 4 hops. The horizontal axis shows the packet error rate on a link (i.e., the probability that a packet is received in error on a link). Note that it is a "packet" error rate, not a "bit" error rate. M_1 and M_3 are the parameters, and their values are indicated by a tuple (M_1, M_3) in this figure. It is apparent that "no error recovery scheme $(M_1 = M_3 = 1)$" gives the worst loss probability. The loss probability improves with the increase in the values of M_1 and M_3. For instance, when the packet error rate on a link p_1 is 10^{-6}, which corresponds to 10^{-9} bit error rate, a maximum of two transmissions for both a link-by-link scheme (indicated by the line (2, 1) in Figure 3.5) and an edge-to-edge scheme (indicated by the line (1, 2)) gives less than 10^{-10} packet loss probability across a network.

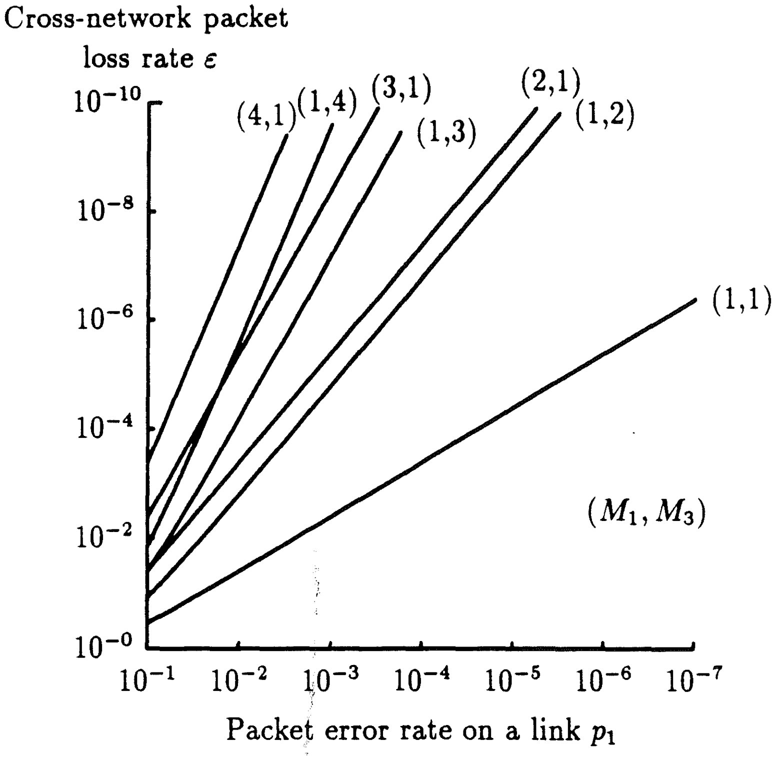


Figure 3.5: Cross-Network Packet Loss Rate (4 hops)

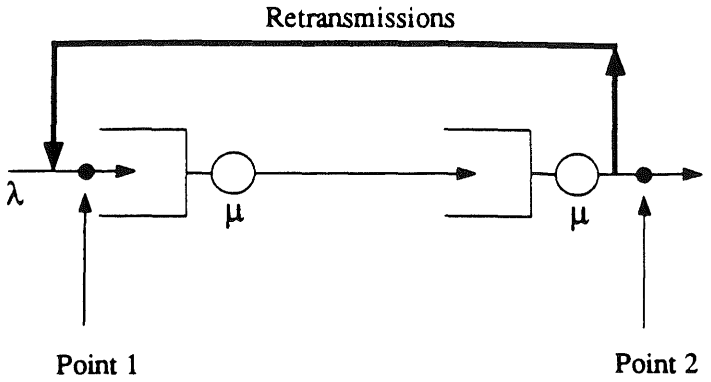


Figure 3.6: Simulation Model

Since optical fibers can easily achieve a packet error rate of 10^{-6} (or equivalently, a bit error rate of 10^{-9}) on a link, and since 10^{-10} packet loss probability across the network is small enough to satisfy the requirements for broadband networks (i.e., the cross-network packet loss rate of less than 10^{-9}), a maximum of two transmissions for both a link-by-link scheme and an edge-to-edge scheme ($M_1 = M_3 = 2$) is sufficient in high-speed networks.

3.4.2 Accuracy of the Poisson Assumption

In the analysis, it is assumed that the aggregated arrivals of new and retransmitted packets follow a Poisson process. (See subsection 3.4.1 for the exact statement of this assumption.) An intuitive justification was given in subsection 3.3.1. In this subsection, we verify the accuracy of this assumption through simulations.

Figure 3.6 represents the queueing model used in the simulations for both link-by-link and edge-to-edge schemes. This one-hope network is simulated allowing the maximum of two transmissions per packet (initial transmission and one retransmission). Throughout this subsection, an average packet transmission time

($P/V \approx 1.666 \times 10^{-6}$ sec) is used as a unit time. In our simulations, the protocol-processing time at each queue is assumed to follow an exponential distribution with the average (denoted by $\frac{1}{\mu}$) equal to 5. Arrivals of new packets (excluding the packets due to retransmissions) to the first queue are assumed to follow a Poisson process. It is also assumed that a link propagation delay is 150. This corresponds to an inter-queue distance of 50 Km. The time-out period is assumed to be 420 times the average packet transmission time, and this time-out period includes the round-trip propagation delay, the queueing delay and the processing time. Refer to Figure 3.2 (b). (This large time-out period helps to reduce the number of unnecessary time-outs.) The parameter values used in our simulations correspond to the following parameter values in the analysis: $\frac{1}{\mu_{1,t}} = \frac{1}{\mu_{1,e}} = 5$, $\mu_2 = \mu_{3,t} = \infty$, and $\tau_1 = 420$ for the link-by-link scheme, and $\frac{1}{\mu_{3,t}} = \frac{1}{\mu_{3,e}} = 5$, $\mu_2 = \mu_{1,t} = \infty$, and $\tau_3 = 420$ for the edge-to-edge scheme.

In our simulations, the Poisson assumption is tested at two points, Point 1 and Point 2 in Figure 3.6. At Point 1, the aggregated arrivals of new and retransmitted packets are tested. At Point 2, the departure process after packet discarding takes place is tested. Chi-Square Goodness-of-Fit Test [ALLE78] is performed at these two points. In addition, the average end-to-end packet transfer delays are obtained through simulations and compared with the analytical results. Both Selective-Repeat and Go-Back-N retransmission procedures are examined. Tables 3.2 through 3.4 show the results of Chi-Square Goodness-of-Fit Test. Figures 3.7 through 3.9 show the comparison of the average packet transfer delays obtained through simulations and through the analysis.

Table 3.2 show the Goodness-of-Fit Test results for the Selective-Repeat retransmission procedure. A packet error rate of 0.05 is assumed. This packet error rate corresponds to 5×10^{-5} bit error rate on a link. Our test shows that, with the 5%

Test Point / λ/μ	Point 1	Point 2
0.2	Poisson	Poisson
0.4	Poisson	Poisson
0.6	Poisson	Poisson
0.8	Poisson	Poisson

Level of significance = 5%
Packet error rate = 0.05

Table 3.2: Results of Chi-Square Goodness-of-Fit Test (Selective-Repeat)

level of significance, both the arrival pattern (at point 1) and the departure pattern (at point 2) appear to be Poisson over a wide range of traffic intensities ($\frac{\lambda}{\mu}$).

Figure 3.7 shows the average packet transfer delay for the same model used in Table 3.2. The Selective-Repeat procedure is assumed, and the packet error rate is assumed to be 0.05. The horizontal axis shows the traffic intensity $\frac{\lambda}{\mu}$. The simulation results match very closely with the analytical results for a wide range of traffic intensities. This further verifies the accuracy of our approximation for the Selective-Repeat procedure.

In Table 3.2 and Figure 3.7, the accuracy of the Poisson assumption is established for a network with 0.05 packet error rate on a link. This Poisson assumption will certainly provide a good approximation for a network with smaller error rates, since with smaller error rates, retransmissions and losses occur less often.

In Tables 3.3, 3.4 and Figures 3.8, 3.9, the Go-Back-N procedure is examined. In Table 3.3 and Figure 3.8, the packet error rate is assumed to be 10^{-3}. This

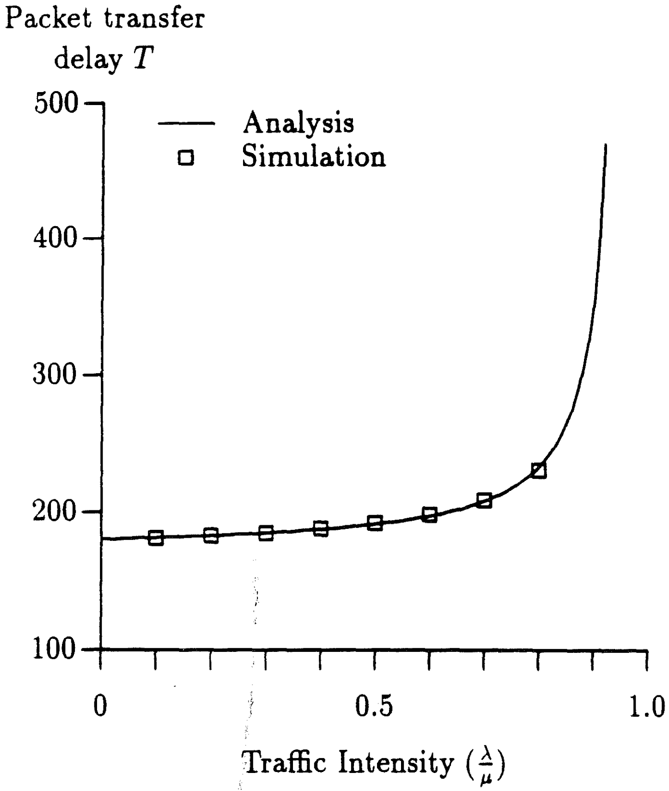


Figure 3.7: Avg. End-to-End Packet Transfer Delay (Selective-Repeat, $p_1 = 0.05$)

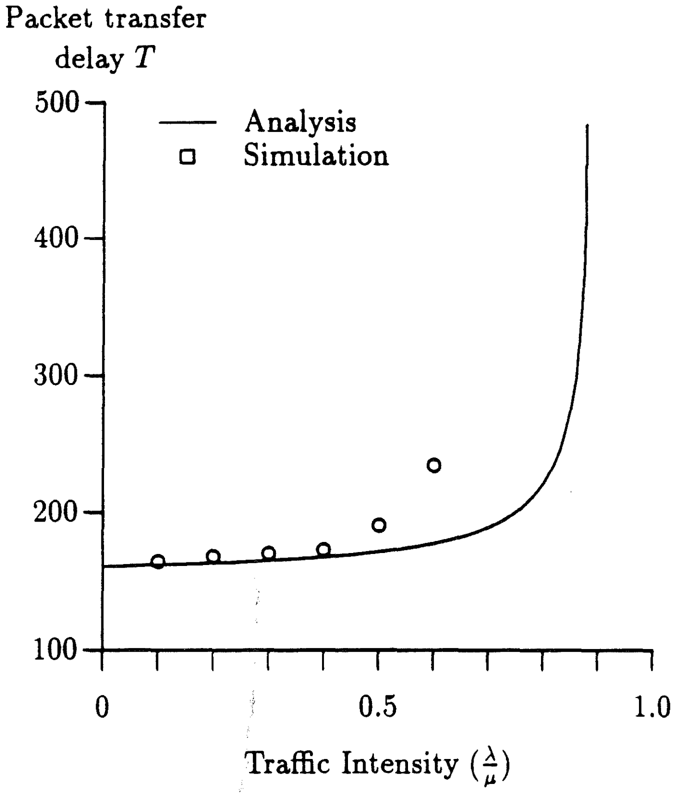


Figure 3.8: Avg. End-to-End Packet Transfer Delay (Go-Back-N, $p_1 = 10^{-3}$)

Test Point λ/μ	Point 1	Point 2
0.1	Poisson	Poisson
0.2	Poisson	Poisson
0.3	Poisson	Poisson
0.4	Poisson	Poisson
0.5	Not Poisson	Not Poisson

Level of significance = 5%
Packet error rate = 10^{-3}

Table 3.3: Results of Chi-Square Goodness-of-Fit Test (Go-Back-N)

packet error rate corresponds to 10^{-6} bit error rate on a link. Table 3.3 shows that the Poisson assumption holds only when the traffic intensity ($\frac{\lambda}{\mu}$) is low (up to, but not including 0.5). Figure 3.8 also shows that our approximation is accurate up to the traffic intensity of approximately 0.5. Beyond this point, simulations yield larger delays than the analysis and reach saturation sooner. This discrepancy between the analysis and simulations is caused by the correlation in the arrival process due to retransmissions. (In the analysis, this correlation is not considered.) However, as we will see later (in Figure 3.15), the range of traffic intensities where the edge-to-edge scheme provides smaller delay than the link-by-link scheme (from 0 up to approximately 0.3) falls within the range where our approximation provides accurate results (from 0 up to 0.5). Therefore, our approximation for Go-Back-N is useful even when the packet error rate is relatively large.

Test Point λ/μ	Point 1	Point 2
0.2	Poisson	Poisson
0.4	Poisson	Poisson
0.6	Poisson	Poisson
0.8	Poisson	Poisson

Level of significance = 5%
Packet error rate = 10^{-6}

Table 3.4: Results of Chi-Square Goodness-of-Fit Test (Go-Back-N)

In Table 3.4 and Figure 3.9, the Go-Back-N procedure and the packet error rate of 10^{-6} are assumed. This packet error rate corresponds to 10^{-9} bit error rate on a link. Both Table 3.4 and Figure 3.9 show that our approximation is accurate for a wide range of traffic intensities.

In summary, for the Selective-Repeat procedure, our analysis provides an accurate approximation for a wide range of traffic intensities and packet error rates. For the Go-Back-N procedure, when the packet error rate is large, our approximation yields accurate results only for a limited range of traffic intensities. However, when the packet error rate is small, our approximation is accurate for a wide range of traffic intensities for the Go-Back-N procedure. The use of optical fibers can provide a low error rate on a link, and thus, our analysis provides an accurate approximation for the networking environment of practical interests.

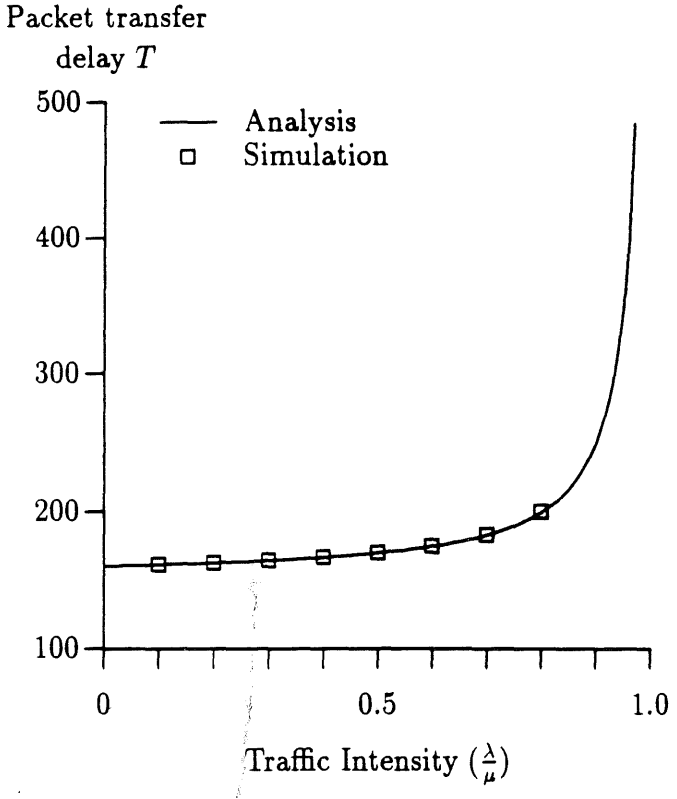


Figure 3.9: Avg. End-to-End Packet Transfer Delay (Go-Back-N, $p_1 = 10^{-6}$)

3.4.3 Packet Transfer Delay

In the edge-to-edge scheme hop-by-hop error checking and retransmissions are eliminated, and thus, the edge-to-edge scheme requires less protocol-processing overhead than the link-by-link scheme. On the other hand, in the edge-to-edge scheme, retransmissions take place between the source and the destination, not between two adjacent nodes. Thus, the edge-to-edge scheme requires larger retransmission delay overhead. In this subsection, numerical results are presented to show this performance trade-off between the link-by-link and the edge-to-edge schemes.

Throughout this subsection, the network model in Figure 3.1 is used, and an average packet transmission time ($P/V \approx 1.666 \times 10^{-6}$ sec) is used as a unit time. As in subsection 3.4.2, a link propagation delay D_{prop} is assumed to be 150 (i.e., internode distance of 50 Km), and the link-by-link time-out period τ_1 is set to 420. The edge-to-edge time-out period τ_3 is set to 1680, four times the link-by-link time-out period. (The edge-to-edge time-out period is set to four times the link-by-link time-out period since a 4-hop network is used in this subsection.) The average protocol-processing time at each layer is assumed to be the same (i.e., $\frac{1}{\mu_{3,t}} = \frac{1}{\mu_{3,e}} = \frac{1}{\mu_2} = \frac{1}{\mu_{1,t}} = \frac{1}{\mu_{1e}}$) and is 5.

In Figure 3.10 through Figure 3.14, the Selective-Repeat procedure is assumed in both edge-to-edge and link-by-link schemes. Figures 3.10, 3.11 and 3.12 compare the average packet transfer delay T in the link-by-link scheme and in the edge-to-edge scheme for various packet error rates. The horizontal axis shows the traffic intensity $\frac{\lambda_1}{\mu_{3,t}}$ at PT-layer at the source node. Both the analytical results and the simulation results are shown in these figures. The simulation results match very closely with the analytical results. This further validates the assumptions employed in our analysis.

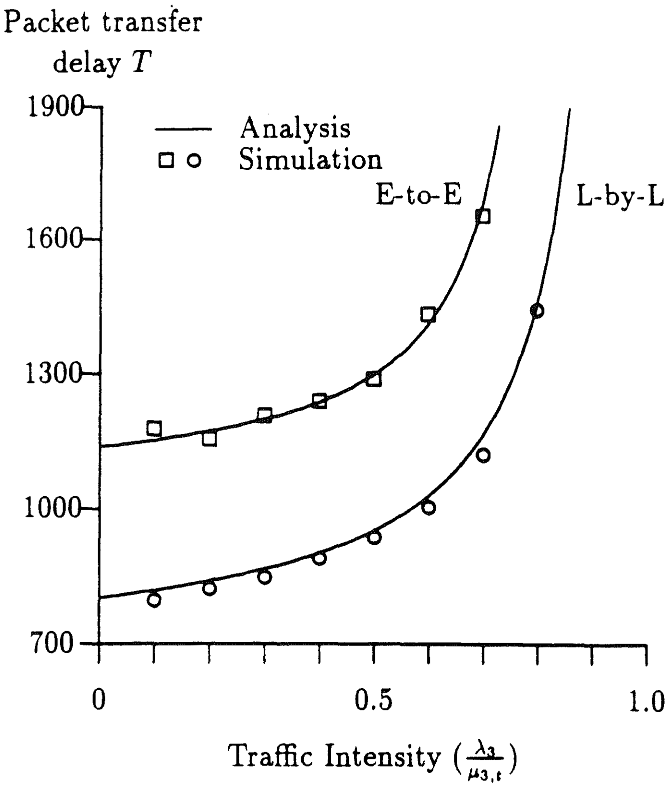


Figure 3.10: Avg. End-to-End Packet Transfer Delay (Selective-Repeat, $p_1 = 0.05$)

In Figure 3.10, the packet error rate p_1 of 0.05 is assumed. In this figure, the edge-to-edge scheme provides the larger delay for all the values of the traffic intensity. This is due to the following reason: when the packet error rate is large, the drawback of the edge-to-edge scheme (i.e., increased retransmission delay overhead) outweighs its benefit (i.e., reduced processing time), and thus, the edge-to-edge scheme gives the larger delay than the link-by-link scheme.

In Figure 3.11, the packet error rate on a link p_1 is decreased to 10^{-3}. This packet error rate corresponds to the bit error rate of 10^{-6}, a typical bit error rate on existing networks. This figure shows that the edge-to-edge scheme provides the smaller delay for all ranges of the traffic intensity. This is because, as the packet error

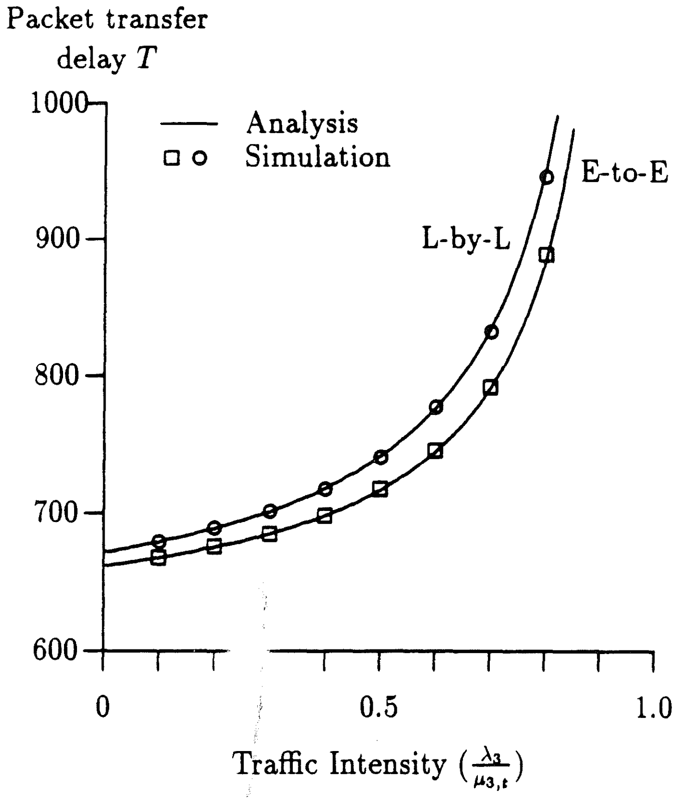


Figure 3.11: Avg. End-End Packet Transfer Delay (Selective-Repeat, $p_1 = 10^{-3}$)

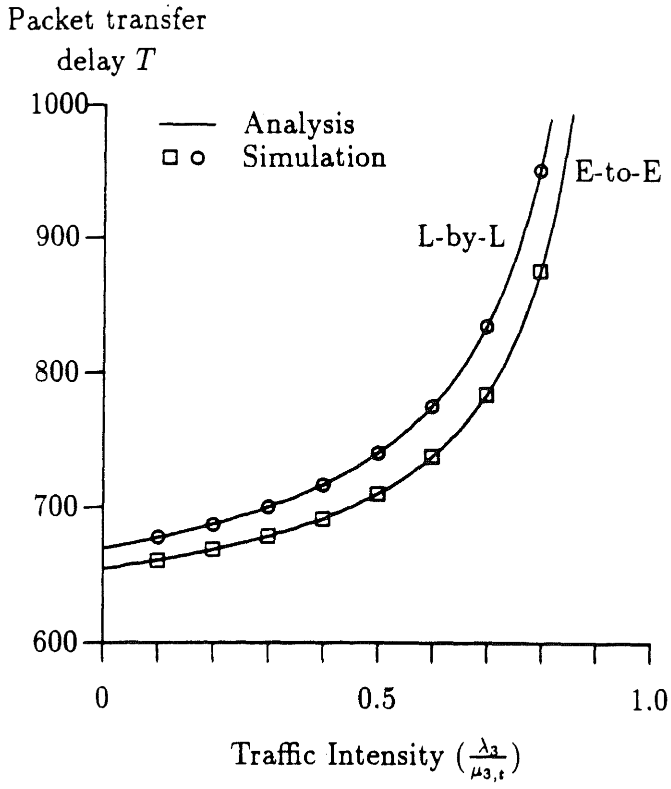


Figure 3.12: Avg. End-to-End Packet Transfer Delay (Selective-Repeat, $p_1 = 10^{-6}$)

rate decreases, the advantage of the reduced processing overhead in the edge-to-edge scheme outweighs its drawback of increased retransmission delay overhead.

In Figure 3.12, the packet error rate on a link p_1 is further decreased to 10^{-6}. This packet error rate corresponds to 10^{-9} bit error rate, a typical bit error rate on a high-speed network. Figure 3.12 is very similar to Figure 3.11. This is because the number of retransmissions in both figures is very small, leading to the similar delay performance.

Figures 3.13 and 3.14 assume the Selective-Repeat procedure and illustrates an optimal error recovery scheme as a function of a packet error rate p_1 and the traffic intensity. The area "[E-to-E]" shows the area where the edge-to-edge scheme provides the smaller delay. The area "[L-by-L]" shows the area where the link-by-link

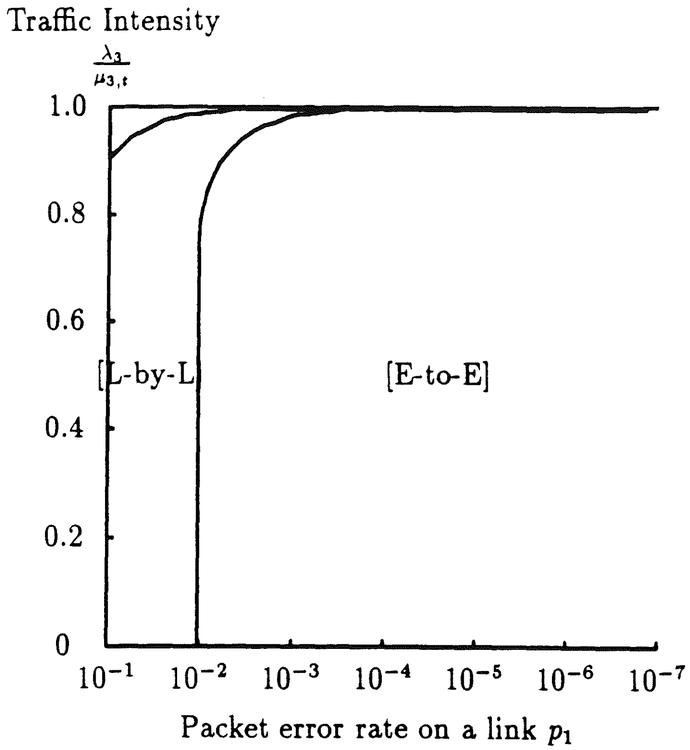


Figure 3.13: Optimal Error Recovery Scheme (Selective-Repeat)

scheme provides the smaller delay. The area not labeled (i.e., the area above [L-by-L]) is the area where a network becomes saturated. In Figure 3.13, as in Figures 3.10, 3.11 and 3.12, the average protocol processing time is assumed to be 5, i.e., $\frac{1}{\mu_{3,t}} = \frac{1}{\mu_{3,e}} = \frac{1}{\mu_2} = \frac{1}{\mu_{1,t}} = \frac{1}{\mu_{1e}} = 5$. In Figure 3.14, the average protocol-processing time is assumed to be almost negligible, namely, 0.001 times the average packet transmission time.

In Figure 3.13, the vertical line represents the traffic intensity $\frac{\lambda_3}{\mu_{3,t}}$. If the packet error rate p_1 is slightly larger than 10^{-2}, the link-by-link scheme yields a smaller delay than the edge-to-edge scheme for almost entire ranges of traffic intensity. (For extremely high traffic intensities close to 1, a network becomes saturated.) If p_1 is 10^{-3}, the edge-to-edge scheme almost always yields a smaller delay than the link-by-link scheme. From this figure, it can be seen that with the Selective-Repeat procedure,

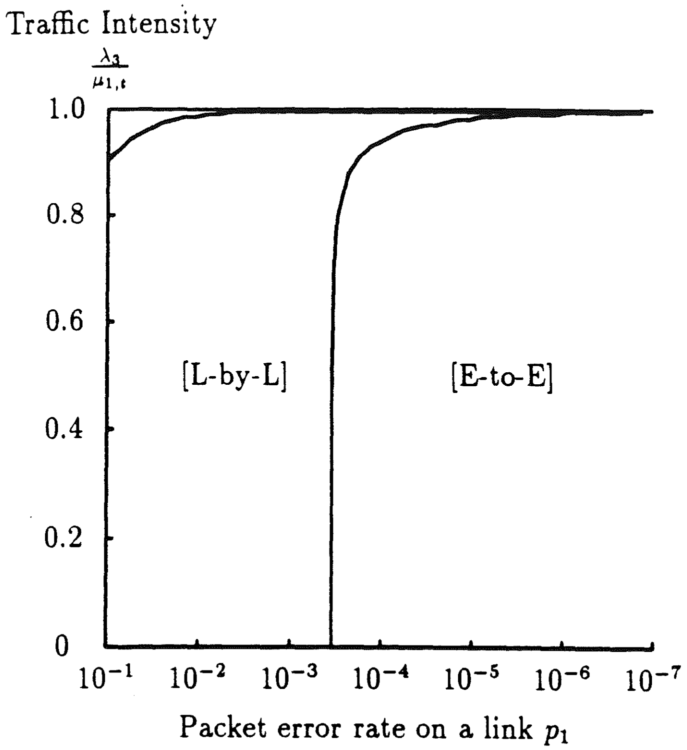


Figure 3.14: Optimal Error Recovery Scheme with a Decreased Processing Time (Selective-Repeat)

the edge-to-edge scheme provides a smaller delay than the link-by-link scheme for a very wide traffic range when the packet error rate is smaller than 10^{-2}. This result suggests the superiority of the edge-to-edge scheme to the link-by-link scheme in high-speed networks where the packet error rate is expected to be very small.

Figure 3.14 shows the effect of the decreased processing time on the optimal error recovery scheme. In this figure, the protocol-processing time is assumed to be almost negligible, and we assume $\frac{1}{\mu_{3,t}} = \frac{1}{\mu_{3,e}} = \frac{1}{\mu_2} = \frac{1}{\mu_{1,e}} = 0.001$, and $\frac{1}{\mu_{1,t}} = 1$. The vertical axis shows the traffic intensity $\frac{\lambda_3}{\mu_{1,t}}$. $\mu_{1,t}$ is used instead of $\mu_{3,t}$ since the packet transmission time is the bottleneck in this figure. Note that in the previous figures (Figure 3.10 through Figure 3.13), the protocol-processing time is assumed to be 5 times the average packet transmission time. By comparing Figures 3.13 and

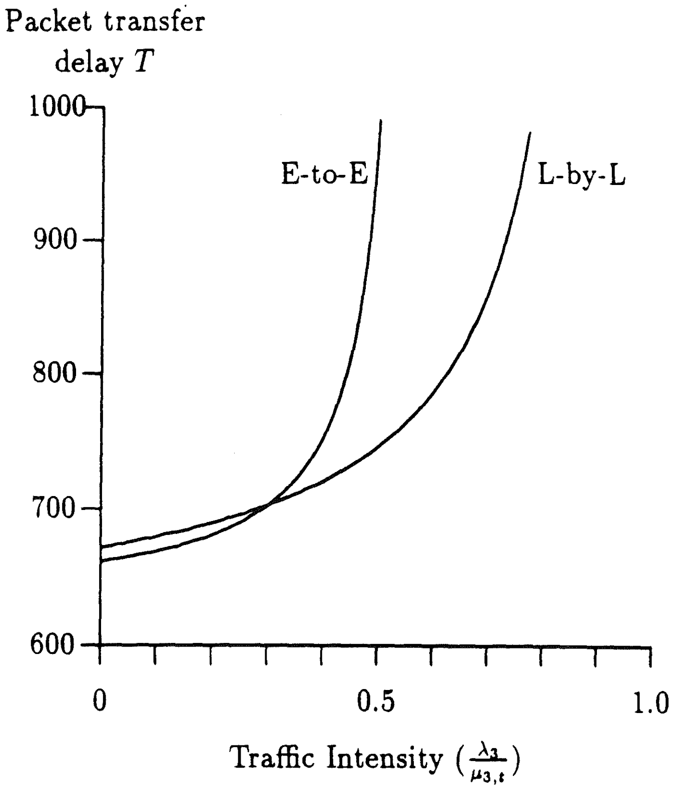


Figure 3.15: Avg. End-to-End Packet Transfer Delay (Go-Back-N, $p_1 = 10^{-3}$)

3.14 it can be seen that when processing time is decreased, the area where the link-by-link scheme gives the smaller delay becomes wider. This is because the benefit of reduced protocol-processing overhead in the edge-to-edge scheme decreases when the processing time at each node becomes smaller.

In Figure 3.15 through Figure 3.18, the Go-Back-N procedure is assumed in both edge-to-edge and link-by-link schemes. Figure 3.15 compares the average end-to-end packet transfer delay T in the link-by-link scheme and in the edge-to-edge scheme. The packet error rate p_1 of 10^{-3} is assumed. In this figure, the edge-to-edge scheme provides the smaller average transfer delay when the traffic intensity is low (up to the traffic intensity of approximately 0.3). [2] As the traffic intensity increases, however, the edge-to-edge scheme gives larger delay than the link-by-link scheme and reaches the saturation sooner than the link-by-link scheme. This is due to the trade-off between the protocol-processing overhead and the time required to recover from an error. When the traffic intensity is low, and thus, the number of erred packets is small, the drawback of the edge-to-edge scheme (i.e., increased retransmission delay overhead) is outweighed by the benefit of the reduced processing overhead.

In Figure 3.16, the packet error rate on a link p_1 is decreased to 10^{-6}, which represents the packet error rate in high-speed network environments. In this figure, the edge-to-edge scheme provides the smaller delay for all ranges of the traffic intensity. This is due to the following reason: when the error rate is extremely small, the number of erred packets is also small even when the traffic intensity is high. Therefore, the drawback of the edge-to-edge scheme (i.e., increased retransmission delay overhead) is outweighed by its benefit (i.e., reduced processing overhead), and thus, the edge-to-edge scheme gives the smaller delay. From Figures 3.15 and 3.16 it can be concluded that as the packet error rate decreases, the edge-to-edge scheme provides the smaller average delay than the link-by-link scheme for a wide traffic range.

Figure 3.17 illustrates an optimal error recovery scheme to give the smallest average transfer delay for a given packet error rate p_1 and the traffic intensity $\frac{\lambda_3}{\mu_{3,t}}$. For instance, if $p_1 = 10^{-3}$, the edge-to-edge scheme yields the smaller delay than the link-by-link scheme in the traffic range $0 < \frac{\lambda_3}{\mu_{3,t}} < 0.3$. For the packet error probability

[2] As shown in Figure 3.8 in subsection 3.4.2, when $p_1 = 10^{-3}$, our analytic results closely match with the simulation results up to the traffic intensity of 0.5. In Figure 3.15, the lines for the edge-to-edge and the link-by-link schemes cross at the traffic intensity of approximately 0.3, within the traffic range where our analysis and simulations match well. Thus, the discussion on Figure 3.15 presented here still holds.

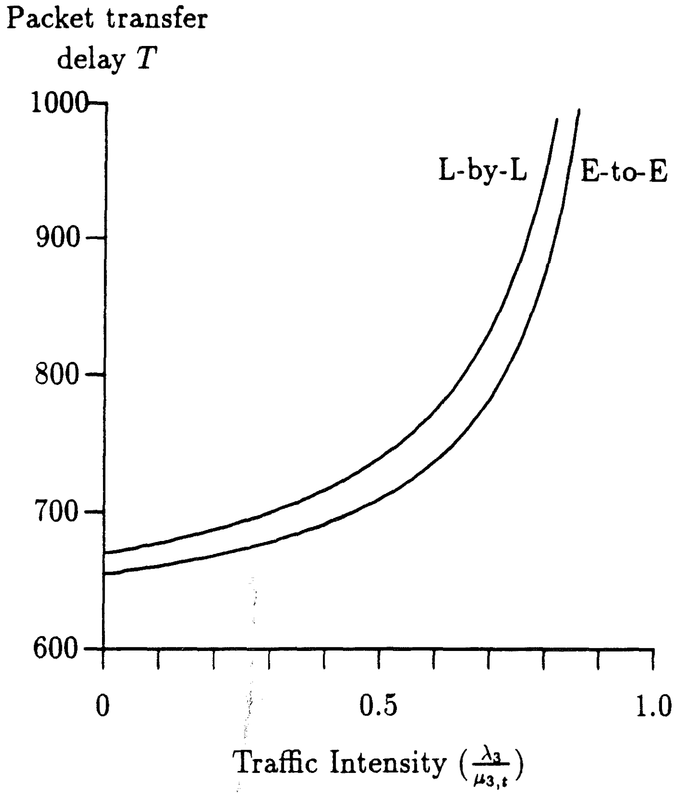


Figure 3.16: Avg. End-to-End Packet Transfer Delay (Go-Back-N, $p_1 = 10^{-6}$)

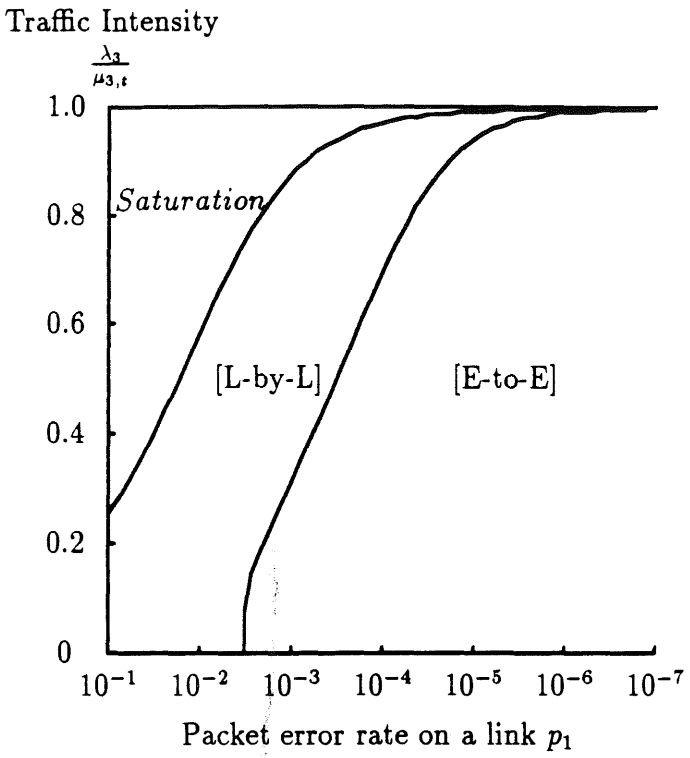


Figure 3.17: Optimal Error Recovery Scheme (Go-Back-N)

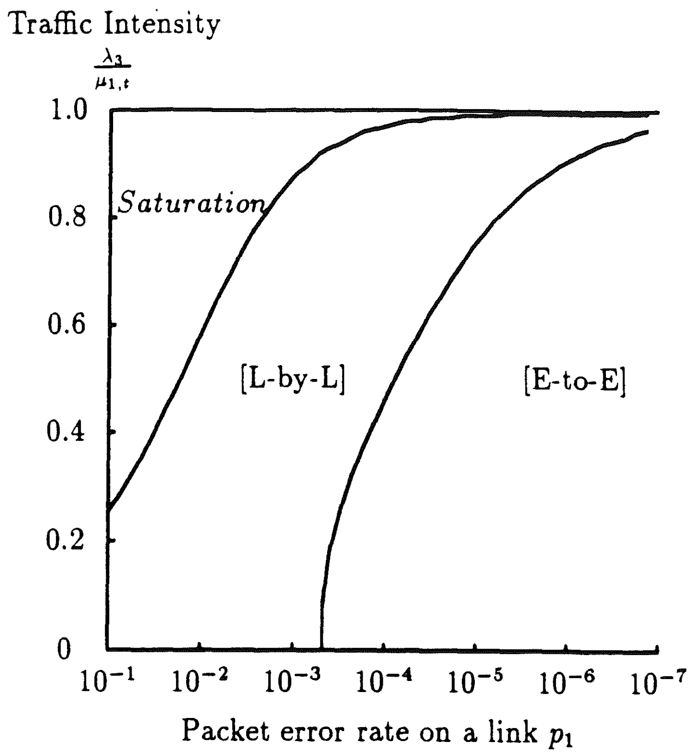


Figure 3.18: Optimal Error Recovery Scheme with a Decreased Processing Time (Go-Back-N)

of 10^{-6}, the edge-to-edge scheme almost always (except for the extremely high traffic intensity close to 1) yields the smaller delay than the link-by-link scheme. Figure 3.17 shows that as the packet error probability p_1 on a link decreases, the edge-to-edge scheme yields the smaller delay for the wider traffic range, and eventually, the edge-to-edge scheme becomes superior to the link-by-link scheme for all the traffic range. Since optical fibers can easily achieve low packet error rates (i.e., 10^{-6} or smaller), we can conclude that the edge-to-edge scheme is superior to the link-by-link scheme in high-speed networks.

Figure 3.18 shows the effect of the decreased processing time on the optimal error recovery scheme. In this figure, the protocol-processing time is assumed to be almost negligible (i.e., 0.001 times the average packet transmission time). As in

Retransmission Procedure Error Probability	Go-Back-N	Selective-Repeat
10^{-6}	89	>1000
10^{-5}	28	>1000
10^{-4}	8	236
10^{-3}	1	24

Table 3.5: Number of Hops Required to Make Link-by-Link Perform As Well As Edge-to-Edge

the Selective-Repeat procedure case, when the processing time is decreased, the area where the link-by-link scheme gives the smaller delay becomes wider.

Finally, the effect of the number of hops l between the source and the destination on the average transfer delay is examined. For each of the retransmission procedures, Table 3.5 shows the number of hops at which the performance of the link-by-link scheme surpasses that of the edge-to-edge scheme (crossover point). The packet error probability on a link p_1 is the parameter in this table. The traffic intensity $\frac{\lambda_3}{\mu_{3,t}}$ at PT-layer of the source node is fixed to 0.5. For instance, when p_1 is 10^{-6}, $\frac{\lambda_3}{\mu_{3,t}}$ is 0.5, and the Go-Back-N procedure is used, the edge-to-edge scheme provides the smaller transfer delay than the link-by-link scheme for a network with less than 89 hops. For a network with 89 hops or more, the link-by-link scheme provides the smaller transfer delay. This result implies that as the number of hops increases, the performance of the link-by-link scheme approaches that of the edge-to-edge scheme, and eventually, the former will surpass the latter. This is because the inefficiency of the edge-to-edge scheme increases as the number of hops increases; as the number of hops increases, retransmissions between the source and the destination require longer time.

From Table 3.5, it can also be seen that the crossover in the Selective-Repeat procedure happens at a larger value of l than in the Go-Back-N procedure. This is because the number of packets needed to be retransmitted in Selective-Repeat is much smaller than that in Go-Back-N. Therefore, with Selective-Repeat, the edge-to-edge scheme performs better than the link-by-link scheme in the wider range of parameter values. Table 3.5 also indicates that as the error probability increases, the crossover happens at a smaller value of l. This is due to the following reason: as the error probability increases, more number of packets are retransmitted. Since one retransmission takes longer time in the edge-to-edge scheme than in the link-by-link scheme, the performance of the edge-to-edge scheme becomes significantly worse. Therefore, the crossover happens at a smaller value of l when the error probability increases.

3.5 Chapter Summary

In this chapter, we investigated an edge-to-edge error recovery scheme for a high-speed packet switched network and obtained both the packet transfer delay and the packet loss probability across a network. The performance of an edge-to-edge scheme was compared with that of a link-by-link scheme, which is an error recovery scheme used in traditional networks. Through analysis, the effects of protocol-processing overhead on the performance of error recovery schemes were investigated. System parameters such as the channel bit-error rate and the number of hops between the source and the destination were varied to investigate how these parameters affect the network performance. The results showed that in high-speed network environments, where the channel speed is very high and the error probability on a link is very small, the edge-to-edge scheme provides a smaller transfer delay than the link-by-link scheme for both Go-Back-N and Selective-Repeat retransmission procedures.

There are some possible improvements that can be made to the work presented in this chapter. For instance, the analysis was developed assuming constant propagation delay (D_{prop}) and constant time-out periods (τ_1 and τ_3) in this chapter. Our current analysis can be expanded to accommodate variable propagation delays and variable time-out periods, with slightly more complicated notation.

Another possible improvement is explained below. In this chapter, we mainly focused on retransmissions due to bit errors. Errors caused by packet loss due to buffer overflow can also trigger retransmissions. In our analysis, the retransmissions due to buffer overflow were not explicitly considered. However, with minor changes, the current analysis can still accommodate this factor.

Let us assume that the errors do not occur in a bursty fashion. This is a very common assumption made in a number of papers (see, for instance, [BKTV88, LAM76, IP80, KUHL83, BS90a, NL90]). This assumption makes it possible to use the model assumed in this chapter to investigate the effects of retransmissions due to buffer overflow. Under this assumption, buffer overflow occurs randomly. Therefore, errors (or retransmissions) due to buffer overflow can be easily accommodated by simply increasing the bit error rate on a link.

In order to be more specific, let us say that p is the packet error rate on a link and that z_i is the blocking probability at node i. z_i can be easily computed from the analysis of the M/M/1/K queue [ALLE78]. Then, by using $new_p = z_i + p - z_ip$ as a probability for retransmissions, and by assuming unacknowledged packets (i.e., packets waiting to be acknowledged) are stored in a separate buffer and do not contribute to the buffer overflow, the analysis in this chapter as it stands can take into account the packet loss due to buffer overflow.

Chapter 4

Congestion Control for Multimedia Traffic

In this chapter, we investigate an effective control scheme for multimedia traffic: one of the two key research issues described in Chapter 1. As described in the previous chapters, high-speed networks are expected to support diverse applications. Different applications have different traffic characteristics and require different grades of service. Many architectures proposed for high-speed networks, such as ATM and IBM's PARIS [CGGK90], are based on packet switching and explicitly permit packet loss in order to gain bandwidth efficiency. Therefore, it is important to predict whether a network can provide a required GOS (i.e., an acceptable level of packet loss) for each of the service classes on a network.

In this chapter, we study a queueing system where heterogeneous traffic streams (sessions) are multiplexed and investigate how the heterogeneity of the arrival streams, especially the varying level of burstiness, affects packet loss in individual streams. We consider the class of Markov Modulated Arrival (MMA) streams both in continuous time (a Markov Modulated Poisson Process or MMPP) [HL86] and in discrete-time (a Markov Modulated Bernoulli Process or MMBP) and present an exact analysis of individual packet loss for MMA streams.

We also analyze packet loss when a packet discarding control scheme is applied as a congestion control mechanism. Several buffer control schemes have been proposed to alleviate the problem of packet loss. For example, low priority packets may be discarded when the buffer is filled to a certain level. In these cases it is especially important to predict packet losses for individual streams with different priorities. We study a previously proposed packet discarding scheme [BCS90] and derive expressions for individual packet loss.

The important contribution of our work is twofold – the study of the impact of burstiness of traffic streams on the individual packet loss probabilities and the study of the effectiveness of priority packet discarding. We present several numerical results describing individual packet loss probabilities when bursty streams are multiplexed with non-bursty streams. We introduce the concept of self-loss for a single stream, the packet loss incurred when a stream is multiplexed with itself, and show how the self-loss of bursty and non-bursty streams may be used to understand the effects of multiplexing heterogeneous arrival streams on individual packet loss. We also present several numerical results illustrating the effectiveness of priority packet discarding. The effects of burstiness, offered load of high priority stream and offered load of low priority stream on the effectiveness of priority packet discarding are also investigated through numerical examples.

Most analytical approaches in the past study the packet loss incurred when several identical arrival streams are multiplexed at a single buffer [AMS82, HL86, NKT91] and thus fail to adequately address the issue of the heterogeneity of the arrival streams. In this chapter, we obtain individual packet loss for both continuous and discrete-time cases, as well as when a buffer control scheme (a priority packet discarding scheme) is in effect. Most of the past research on priority packet discarding (e.g., [BCS90, LP90, KRON90, PF91]) is limited to a Poisson/Bernoulli arrival assumption. In [LI89a], a similar control scheme is analyzed for M/PH/1/N and PH/M/1/N queueing systems. In [LI89a], however, no analysis is presented for the discrete-time case, which introduce the extra complication of simultaneous arrivals.

In this chapter, we follow the stochastic integral approach in [RS], a method of independent interest, to derive our individual packet loss expressions. In doing so, we re-derive expressions for the continuous-time case presented without proof in [MEIE89]. In [MEIE89], the emphasis was on analyzing parcel overflow processes using a two-state MMPP approximation for a multistate MMPP. Numerical results presented therein focused on the accuracy of the approximation.

The rest of the chapter is organized as follows: in Sections 4.1 and 4.2, individual packet loss probabilities are derived for continuous and discrete-time cases when two arrival streams are present. In Section 4.3, our analysis is extended to accommodate a priority packet discarding scheme. Both continuous and discrete-time cases are considered. In Section 4.4, we present several interesting numerical results. Our analysis is applied to investigate the effects of individual traffic characteristics and traffic mix on the individual packet loss probabilities. The effectiveness of priority packet discarding is also investigated using our analysis presented in Section 4.4. Finally, in Section 4.5, a chapter summary is given. Note that our analyses for 2-stream cases can easily be extended for $N(>2)$ heterogeneous traffic streams. In Appendices B and C, analysis for $N(>2)$ heterogeneous streams with no control is presented for continuous and discrete time, respectively. In Appendix D, the analysis of a priority packet discarding scheme for 2-stream case is extended to $N(>2)$ heterogeneous streams.

4.1 Continuous-Time Case

In this section, individual packet loss probabilities are obtained for the continuous-time case. Each arrival stream is modeled by a 2-state MMPP. Note that a 2-state MMPP is a fairly general process – by selecting appropriate parameter values, a 2-state MMPP can represent a Poisson process (suitable to describe data arrivals) and an Interrupted Poisson Process (suitable to describe On/Off traffic sources such as voice). 2-state MMPPs have been used to represent a superposition of several identical sources [HL86], and thus, each arrival stream in our model can be viewed as a single source or a superposition of multiple identical sources. In this section, for simplicity, it is assumed that two heterogeneous streams (stream A and stream B) are multiplexed. The analysis can easily be extended to a case where $N(> 2)$ heterogeneous input streams are multiplexed and is discussed in Appendix B.

4.1.1 Model and Notations

Consider a single first-come-first-served queue driven by two 2-state MMPP arrival processes. The queue has a finite buffer space whose maximum size is $K - 1$ (packets). Thus, the maximum system size (the maximum buffer size plus the packet being served) is K (packets). Service times of packets from streams A and B are exponentially distributed with rate μ. Packets from each of streams A and B arrive according to a 2-state MMPP. A 2-state MMPP is characterized by two alternating 'driving' states. It is assumed that the duration of each state is exponentially distributed and packet arrivals in each state are Poisson processes with different rates.

The driving states of stream A are labeled 1 and 2; the driving states of stream B are labeled 3 and 4. (Refer to Figure 4.1.) For stream A, the transition rate from

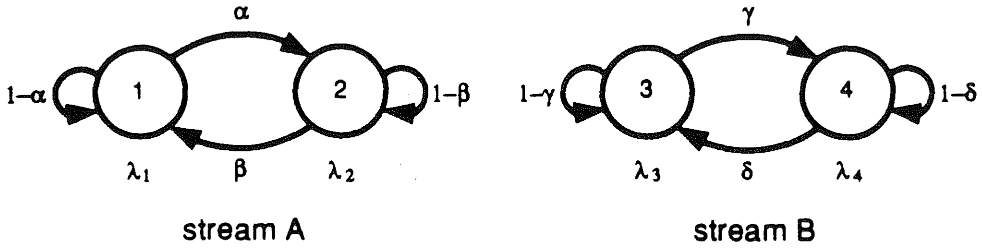


Figure 4.1: 2-State MMPP

state 1 to state 2 is denoted by α, and the transition rate from state 2 to state 1 is denoted by β. For stream B, the transition rate from state 3 to state 4 is denoted by γ, and the transition rate from state 4 to state 3 is denoted by δ. Thus, for stream A, the sojourn times in states 1 and 2 are exponentially distributed with the mean $1/\alpha$ and $1/\beta$, respectively. For stream B, the sojourn times in states 3 and 4 are exponentially distributed with the mean $1/\gamma$ and $1/\delta$, respectively. The generation of packets when the MMPP is in state i follows a Poisson process with rate λ_i. Thus, when stream A is in state i $(i = 1, 2)$ and stream B is in state j $(j = 3, 4)$, the aggregate arrival rate to the queueing system is $\lambda_i + \lambda_j$.

Define $Y_A(t)$ and $Y_B(t)$ as the states of stream A and B MMPP's at time t, respectively, i.e., $Y_A(t) = 1$ or 2, and $Y_B(t) = 3$ or 4. For $i = 1, 2$ and $j = 3, 4$, we define the following indicator functions:

$$I_i(t) = \begin{cases} 1, & \text{if } Y_A(t) = i \\ 0, & \text{otherwise} \end{cases}$$

$$I_j(t) = \begin{cases} 1, & \text{if } Y_B(t) = j \\ 0, & \text{otherwise.} \end{cases}$$

$I_i(t)$ $(I_j(t))$ becomes 1, if the state of the MMPP for stream A (B) at time t is i (j). Otherwise it is 0.

Let $Z(t)$ $(0 \leq Z(t) \leq K)$ denote the system state (i.e., the number of packets in the system including both a server and a buffer) at time t. Define the following indicator function for a system state q $(0 \le q \le K)$.

$$U_q(t) = \begin{cases} 1, & \text{if } Z(t-) = q \\ 0, & \text{otherwise.} \end{cases}$$

$U_q(t)$ is 1, if the system state at time $t-$ is q. Otherwise it is 0.

Let $N_A(t)$ and $N_B(t)$ be the cumulative number of arrivals from stream A and from stream B in the time interval $[0, t]$, respectively. Let $N(t)$ be the cumulative number of arrivals in the time interval $[0, t]$. Thus,

$$N(t) = N_A(t) + N_B(t). \tag{4.1}$$

Let $\Lambda_A(t)$ and $\Lambda_B(t)$ denote the *compensators* [BREM81] for the processes $N_A(t)$ and $N_B(t)$, respectively, so that $N_A(t) - \Lambda_A(t)$ and $N_B(t) - \Lambda_B(t)$ are martingales. (See [LS78], pp.239, for the definition of compensators, and see [BREM81], pp.4, for the definition of martingales.) For instance, for the Poisson process, we have the compensator $\Lambda(t) = \lambda t = \int_0^t \lambda ds$, and for the doubly stochastic Poisson process, we have the compensator $\Lambda(t) = \int_0^t \lambda(s) ds$. (See [BREM81]). For our model, the intensity function $\lambda_A(t)$ for stream A and $\lambda_B(t)$ for stream B are given by

$$\lambda_A(t) = I_1(s)\lambda_1 + I_2(s)\lambda_2, \quad \text{and} \quad \lambda_B(t) = I_3(s)\lambda_3 + I_4(s)\lambda_4. \tag{4.2}$$

Thus, the compensators for $N_A(t)$ and $N_B(t)$ become

$$\Lambda_A(t) = \int_0^t (I_1(s)\lambda_1 + I_2(s)\lambda_2) ds, \quad \text{and} \quad \Lambda_B(t) = \int_0^t (I_3(s)\lambda_3 + I_4(s)\lambda_4) ds. \tag{4.3}$$

We define the following limiting probabilities. Let $\pi(i, j, q)$ $(i = 1, 2,\ j = 3, 4,\ 0 \le q \le K)$ be the limiting distribution for the Markov process $\{Y_A(t), Y_B(t), Z(t)\}$. Let $\pi(i, q)$ $(i = 1, 2,\ 0 \le q \le K)$ be the limiting distribution for the Markov process $\{Y_A(t), Z(t)\}$, and $\pi(j, q)$ $(j = 3, 4,\ 0 \le q \le K)$ be the limiting distribution for the Markov process $\{Y_B(t), Z(t)\}$. Note that $\sum_j \pi(i, j, q) = \pi(i, q)$ and $\sum_i \pi(i, j, q) = \pi(j, q)$.

4.1.2 Analysis

In this analysis, we obtain the following probabilities:

1. the long term probability $P_A(q)$ that an arrival from stream A sees the system in state q,
2. the long term probability $P_B(q)$ that an arrival from stream B sees the system in state q, and
3. the long term probability $P(q)$ that an arbitrary arrival sees the system in state q.

Note that $0 \leq q \leq K$. From these probabilities, we can easily obtain the loss proabilities for stream A ($P_{loss}(A)$) and for stream B ($P_{loss}(B)$) by the following:

$$P_{loss}(A) = P_A(K) \quad \text{and} \quad P_{loss}(B) = P_B(K). \tag{4.4}$$

Further, the overall packet loss probability $P_{loss}(O)$ for the aggregated arrival process (i.e., the loss probability of packets, indistinguishing streams A and B) is given by

$$P_{loss}(O) = P(K) \tag{4.5}$$

First, let us calculate the long term probability $P_A(q)$ for an arrival from stream A to see the system state q. We have

$$P_A(q) = \lim_{t\to\infty} \frac{1}{N_A(t)} \int_0^t U_q(s) dN_A(s) = \lim_{t\to\infty} \frac{t}{N_A(t)} \lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s) dN_A(s). \tag{4.6}$$

By noting that the term $\lim_{t\to\infty} \frac{N_A(t)}{t}$ (the inverse of the first term on the right hand side of the above equation) gives the mean arrival rate of stream A, we obtain the following expression:

$$\lim_{t\to\infty} \frac{N_A(t)}{t} = \frac{\lambda_1 \frac{1}{\alpha} + \lambda_2 \frac{1}{\beta}}{\frac{1}{\alpha} + \frac{1}{\beta}} = \frac{\lambda_1 \beta + \lambda_2 \alpha}{\alpha + \beta}. \tag{4.7}$$

To obtain the term $\lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s)dN_A(s)$ in Eq.(4.6), we use the following:

1. The intensity function $\lambda_A(t)$ for stream A is bounded, i.e.,

$$||\lambda_A(t)|| = \sup_{t\geq 0} |\lambda_A(t)| < \infty.$$

2. $U_q(t)$ is a predictable process.
3. $E(\int_0^t |U_q(s)|d|\Lambda_A(s)|) < \infty$, for every $t \in (0, \infty)$.

The three assertions above are easily verified. Note that $N_A(t)$ is, by definition, a doubly stochastic Poisson process [NEUT89]. $\lambda_A(t)(= I_1(t)\lambda_1 + I_2(t)\lambda_2)$ is bounded since it is equal to either λ_1 or λ_2 which are finite. $U_q(t)$ is a predicatable process since it is left continuous. (Proof of this is given in [BREM81], pp.9.) The last condition may be shown as follows:

$$\begin{aligned} E(\int_0^t |U_q(s)|d|\Lambda_A(s)|) &\leq E(\int_0^t d|\Lambda_A(s)|) = E(\int_0^t |(I_1(s)\lambda_1 + I_2(s)\lambda_2|ds) \\ &\leq E(\int_0^t \max(\lambda_1, \lambda_2)ds) \\ &= E(\max(\lambda_1, \lambda_2)s|_0^t) = \max(\lambda_1, \lambda_2) \times t. \end{aligned}$$

Since both λ_1 and λ_2 are finite,

$$E(\int_0^t |U_q(s)|d|\Lambda_A(s)|) < \infty.$$

Now, we use the following theorem [RS] to obtain the term $\lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s)$ $dN_A(s)$ in Eq.(4.6).

Theorem 1: Let $T = [0, \infty)$. For $t \in T$, assume $N_A(t)$ is a doubly stochastic Poisson process with bounded intensity function $\lambda_A(t)$. Define $R(t)$ as the following:

$$R(t) = \int_0^t U_q(s)dN_A(s) - \int_0^t U_q(s)d\Lambda_A(s),$$

where $U_q(t)$ is the indicator function for a system state q and $\Lambda_A(t)$ is a compensator for $N_A(t)$. If $U_q(t)$ is a predictable process satisfying the following condition for every $t > 0$,

$$E(\int_0^t |U_q(s)|d|\Lambda_A(s)|) < \infty$$

then the following equation holds with probability one.

$$\lim_{t\to\infty} \frac{R(t)}{t} = 0 \tag{4.8}$$

For a proof of this theorem, see [RS].

In [RS], it is shown that the stochastic integral $R(t)$ is a martingale; intuitively, $R(t)$ 'hovers' around zero and thus $\lim_{t\to\infty} \frac{R(t)}{t} = 0$. We now apply Theorem 1 to obtain the term $\lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)dN_A(s)$ in Eq.(4.6). We have

$$\begin{aligned}
\lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)dN_A(s) &= \lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)d\Lambda_A(s) \\
&= \lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)(I_1(s)\lambda_1 + I_2(s)\lambda_2)ds \\
&= \lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)I_1(s)\lambda_1 ds + \lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)I_2(s)\lambda_2 ds \\
&= \lambda_1\pi(1,q) + \lambda_2\pi(2,q).
\end{aligned} \tag{4.9}$$

For the last step, we used the fact that $\lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)I_1(s)ds$ ($\lim_{t\to\infty} \frac{1}{t}\int_0^t U_q(s)I_2(s)$ ds) represents the limiting probability that $\{Y_A(t) = 1, Z(t) = q\}$ ($\{Y_A(t) = 2, Z(t) = q\}$), and that it is equal to $\pi(1,q)$ $(\pi(2,q))$. From Eqs.(4.6), (4.7) and (4.9), we have

$$P_A(q) = \frac{(\alpha+\beta)(\lambda_1\pi(1,q) + \lambda_2\pi(2,q))}{\lambda_1\beta + \lambda_2\alpha}. \tag{4.10}$$

Using the same argument for stream B, we obtain

$$P_B(q) = \frac{(\gamma+\delta)(\lambda_3\pi(3,q) + \lambda_4\pi(4,q))}{\lambda_3\delta + \lambda_4\gamma}. \tag{4.11}$$

Next, we compute the long term probability $P(q)$ of an arbitrary arrival seeing the system state q:

$$\begin{aligned} P(q) &= \lim_{t\to\infty} \frac{1}{N(t)} \int_0^t U_q(s) dN(s) \\ &= \lim_{t\to\infty} \frac{t}{N(t)} \lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s) dN(s) \end{aligned} \qquad (4.12)$$

Since $N(t) = N_A(t) + N_B(t)$, we obtain

$$\begin{aligned} \lim_{t\to\infty} \frac{N(t)}{t} &= \lim_{t\to\infty} \frac{N_A(t)}{t} + \lim_{t\to\infty} \frac{N_B(t)}{t} \\ &= \frac{\lambda_1\beta + \lambda_2\alpha}{\alpha+\beta} + \frac{\lambda_3\delta + \lambda_4\gamma}{\gamma+\delta} \end{aligned} \qquad (4.13)$$

For the last step, we used Eq.(4.7). For the second term in the right hand side of Eq.(4.12), we have

$$\begin{aligned} \lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s) dN(s) &= \lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s) d(N_A(s) + N_B(s)) \\ &= \lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s) dN_A(s) + \lim_{t\to\infty} \frac{1}{t} \int_0^t U_q(s) dN_B(s) \\ &= \lambda_1\pi(1,q) + \lambda_2\pi(2,q) + \lambda_3\pi(3,q) + \lambda_4\pi(4,q) \end{aligned} \qquad (4.14)$$

By substituting Eqs.(4.13) and (4.14) into Eq.(4.12), we have

$$P(q) = \frac{(\alpha+\beta)(\gamma+\delta)(\lambda_1\pi(1,q) + \lambda_2\pi(2,q) + \lambda_3\pi(3,q) + \lambda_4\pi(4,q))}{(\lambda_1\beta + \lambda_2\alpha)(\gamma+\delta) + (\lambda_3\delta + \lambda_4\gamma)(\alpha+\beta)}. \qquad (4.15)$$

In order to obtain probabilities $P_A(q)$ (Eq.(4.10)), $P_B(q)$ (Eq.(4.11)) and $P(q)$ (Eq.(4.15)), we need to obtain the limiting probability $\pi(i,j,q)$ for the Markov process $\{Y_A(t), Y_B(t), Z(t)\}$. Once $\pi(i,j,q)$ is obtained, the marginal distributions $\pi(i,q) = \sum_j \pi(i,j,q)$ and $\pi(j,q) = \sum_i \pi(i,j,q)$ can easily be computed.

We use direct numerical methods to compute $\pi(i,j,q)$. We represent the system state by (i,j,q), where i is the state of stream A, j is the state of stream B, and q is the number of packets in the system. For $i, i' \in \{1,2\}$, $j, j' \in \{3,4\}$, and

$q, q' \in \{0, 1, 2, ..., K\}$, the infinitesimal generator [CINL75] $\mathbf{Q}$ for our system is given by

$$\mathbf{Q}_{(i,j,q)\rightarrow(i',j',q')} = \begin{cases} r_{ii'}, & \text{if } i' \neq i, j' = j, q' = q \\ r_{jj'}, & \text{if } i' = i, j' \neq j, q' = q \\ \lambda_i + \lambda_j, & \text{if } i' = i, j' = j, q' = q + 1 \\ \mu, & \text{if } i' = i, j' = j, q' = q - 1 \\ -r_{i\bar{i}} - r_{j\bar{j}} - \lambda_i - \lambda_j, & \text{if } i' = i, j' = j, q' = q = 0 \\ -\mu - r_{i\bar{i}} - r_{j\bar{j}}, & \text{if } i' = i, j' = j, q' = q = K \\ -\mu - r_{i\bar{i}} - r_{j\bar{j}} - \lambda_i - \lambda_j, & \text{if } i' = i, j' = j, q' = q, 0 < q < K \\ 0, & \text{otherwise} \end{cases} \tag{4.16}$$

where r_{ij} is the transition rate of an arrival process from state i to state j, and $\bar{i}$ represents the complementary state of i. $r_{i,j}$ and $\bar{i}$ are given by $r_{12} = \alpha$, $r_{21} = \beta$, $r_{34} = \gamma$, $r_{43} = \delta$, $\bar{1} = 2$, $\bar{2} = 1$, $\bar{3} = 4$, and $\bar{4} = 3$.

Using the above infinitesimal generator, the steady state equation for our system becomes

$$\pi\mathbf{Q} = 0 \tag{4.17}$$

where $\pi = [\pi(1,3,0), \pi(1,4,0), \pi(2,3,0), \pi(2,4,0), \ldots, \pi(2,3,K), \pi(2,4,K)]$. By solving the above set of equations with the condition $\pi\mathbf{e} = 1$, we can easily obtain the steady-state probability $\pi(i, j, q)$.

4.2 Discrete-Time Case

In this section, individual packet loss probabilities are obtained for the discrete-time case. Again, we first consider multiplexing two heterogeneous streams (stream

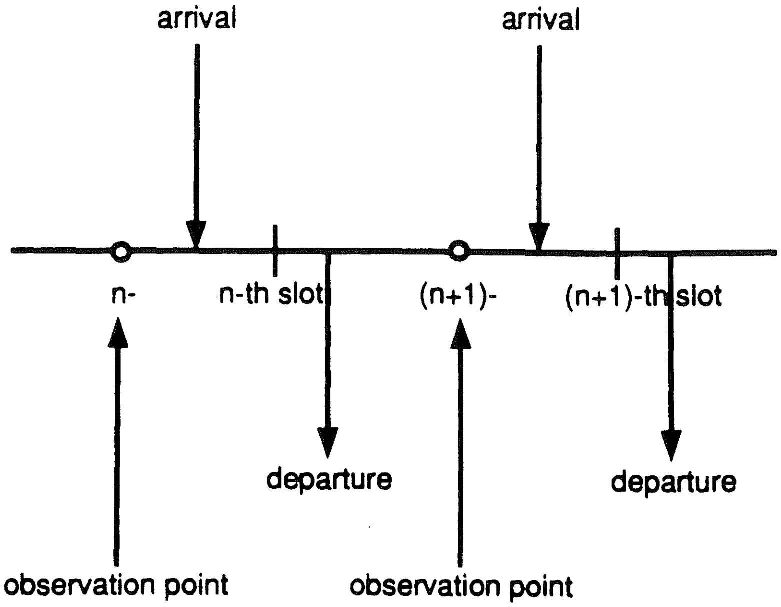


Figure 4.2: Late Arrival System

A and stream B). The extension of the discrete-time analysis for the $N(>2)$ stream case is discussed in Appendix C.

4.2.1 Model and Notations

The same derivation technique used earlier for continuous time is now brought to bear on the discrete-time case. Before we proceed with our analysis, we first decide the order in which arrivals and services take place and the times they occur. Without loss of generality, we assume the late arrival system with immediate access [HUNT83]. In such a system, arrivals occur just prior to the end of a time slot, and the packet in service is ejected from the service facility immediately after the beginning of a time slot (Refer to Figure 4.2). An arriving packet can enter the service facility if it is free, with the possibility of it being ejected almost instantaneously. Note that in this model, packet's service time is counted as the number of slot boundaries from the

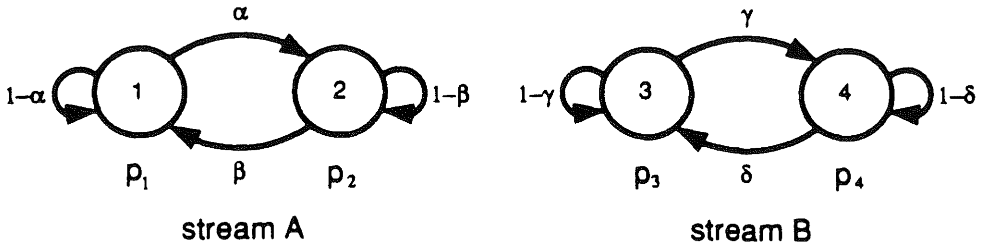


Figure 4.3: 2-State MMPP

entering point to the service facility to the packet departure point. Therefore, even though we allow the arriving packet to be ejected almost instantaneously, its service time is counted as 1, not 0.

Consider a single first-come-first-served queue driven by two 2-state MMBP arrival processes. As in the continuous time case, the 'driving' states of stream A are labeled 1 and 2; the driving states of stream B are labeled 3 and 4 (Refer to Figure 4.3). Without loss of generality, it is assumed that change in the states of the arrival processes occur just prior to the end of a time slot. The sojourn times in states 1 and 2 are geometrically distributed with the mean $1/\alpha$ and $1/\beta$ slots, respectively, and the sojourn times in states 3 and 4 are also geometrically distributed with the mean $1/\gamma$ and $1/\delta$ slots, respectively. Packets arrive according to a Bernoulli process, and the probability of an arrival in a slot is p_i $(0 \leq p_i \leq 1)$ in state i. Service times are geometrically distributed, and the probability of service completion in a slot, provided the server is busy, is s $(0 < s < 1)$ for both stream A and stream B.

Let $n = 0, 1, 2, 3, \ldots$ denote the slot boundary numbers and t continue to denote (continuous) time. Define the step function $\lfloor t \rfloor$, where $\lfloor t \rfloor = n$, if $n \leq t < n+1$. For simplicity, we assume the slot length is equal to a unit time in the system. We observe the system just prior to the end of time slots, i.e., at $n-$ (refer to Figure 4.2).

As in the continuous-time case, we define $Y_A(t)$ and $Y_B(t)$ as the state of the MMBP for stream A and for stream B at time t, respectively. We also let $Z(t)$ $(0 \leq Z(t) \leq K)$ denote the system state at time t. $\pi(i,q)$, $\pi(j,q)$ and $\pi(i,j,q)$ are defined as the limiting probabilities of the Markov processes $\{Y_A(n-), Z(n-)\}$, $\{Y_B(n-), Z(n-)\}$, and $\{Y_A(n-), Y_B(n-), Z(n-)\}$, respectively. Note that $i = 1,2$, $j = 3,4$, and $0 \leq q \leq K$.

In a discrete-time case, a packet arriving at a system whose state is K is lost. Loss of packets can also happen when simultaneous arrivals occur from streams A and B at the system state $K-1$. In such a case, we assume that a packet from stream A is lost with probability P_A, and a packet from stream B is lost with probability $P_B(=1-P_A)$. For a random packet discarding scheme, $P_A = P_B = 0.5$.

In the following analysis, we focus on stream A and obtain its packet loss probability. For each slot n, define the following indicator functions:

- $J(n)$: $J(n) = 1$ iff an arrival has taken place from stream B in the n^{th} slot.
- $V(n)$: $V(n) = 1$ iff in the n^{th} slot, a stream A packet is discarded when a stream B packet arrives (along with a stream A packet) at the system state $K-1$.

Note that, for each n, $V(n)$ is an independent Bernoulli random variable, and thus, $V(1), V(2), \ldots$ is an *iid* sequence.

Using $J(n)$ and $V(n)$, we can obtain the indicator function $U(n)$ for the state in which a stream A packet is discarded, i.e., $U(n) = 1$ iff in the n^{th} slot, stream A is in the state where a stream A packet is discarded. $U(n)$ is given by

$$U(n) = U_K(n) + U_{K-1}(n)J(n)V(n) \tag{4.18}$$

where $U_q(n)$ is the indicator function for the system state q, i.e., $U_q(n) = 1$ *iff* $Z(n-) = q$.

4.2.2 Analysis

First, let us derive the packet loss probability $P_{loss}(A)$ for stream A. From the definition of $P_{loss}(A)$, we have

$$\begin{aligned} P_{loss}(A) &= \lim_{t\to\infty} \frac{1}{N_A(t)} \int_0^t U(s)dN_A(s) \\ &= \lim_{t\to\infty} \frac{t}{N_A(t)} \lim_{t\to\infty} \frac{1}{t} \int_0^t U(s)dN_A(s). \end{aligned} \tag{4.19}$$

Since the term $\lim_{t\to\infty} \frac{N_A(t)}{t}$ represents the mean arrival rate of stream A, it becomes

$$\lim_{t\to\infty} \frac{N_A(t)}{t} = \frac{p_1\beta + p_2\alpha}{\alpha + \beta}. \tag{4.20}$$

In order to obtain the term $\lim_{t\to\infty} \frac{1}{t} \int_0^t U(s)dN_A(s)$, we will use the following manner of writing the compensator for a MMBP:

$$\Lambda_A(t) = \int_0^t (I_1(s)p_1 + I_2(s)p_2)d\lfloor s \rfloor. \tag{4.21}$$

(The following integral and sum are the same: $\int_0^t I_1(s)d\lfloor s \rfloor = \sum_{i=0}^{\lfloor t \rfloor} I_1(i)$.) We may now apply Theorem 1 directly. (Note that 'discreteness' is accounted for via the compensator.)

$$\begin{aligned} \lim_{t\to\infty} \frac{1}{t} \int_0^t U(s)dN_A(s) &= \lim_{t\to\infty} \frac{1}{t} \int_0^t U(s)d\Lambda_A(s) \\ &= \lim_{t\to\infty} \frac{1}{t} \int_0^t U(s)(I_1(s)p_1 + I_2(s)p_2)d\lfloor s \rfloor \\ &= p_1 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U(n)I_1(n) + p_2 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U(n)I_2(n) \\ &= p_1 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_K(n)I_1(n) \end{aligned}$$

$$\begin{aligned}
& +p_1 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_{K-1}(n)J(n)V(n)I_1(n) \\
& +p_2 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_K(n)I_2(n) \\
& +p_2 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_{K-1}(n)J(n)V(n)I_2(n) \\
= \; & p_1\pi(1,K) + p_1 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} V(n) \\
& \cdot \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_{K-1}(n)J(n)I_1(n) + p_2\pi(2,K) \\
& +p_2 \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} V(n) \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_{K-1}(n)J(n)I_2(n).
\end{aligned} \tag{4.22}$$

In the above derivation, we used Eq.(4.18). For the last step, we used the fact that $\lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_K(n)I_1(n)$ ($\lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_K(n)I_2(n)$) represents the limiting probability for $\{Z(n-) = K, Y_A(n-) = 1\}$ ($\{Z(n-) = K, Y_A(n-) = 2\}$), and that it is equal to $\pi(1,K)$ ($\pi(2,K)$).

Note that, since we assume that a packet from stream A is discarded with probability P_A when two arrivals occur in state $K-1$, $P[V(n) = 1] = P_A$. Observe that $\lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} V(n) = P_A$. Also note that $\lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_{K-1}(n)J(n)I_1(n)$ represents the limiting probability that system state is $K-1$, an arrival from stream B occurs, and stream A is in state 1. It is thus equal to $p_3\pi(1,3,K-1)+p_4\pi(1,4,K-1)$. Similarly, $\lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} U_{K-1}(n)J(n)I_2(n)$ is equal to $p_3\pi(2,3,K-1)+p_4\pi(2,4,K-1)$. Then, the above equation (Eq.(4.22)) becomes

$$\begin{aligned}
\lim_{t\to\infty} \frac{1}{t} \int_0^t U(s)dN_A(s) &= p_1\pi(1,K) + P_A p_1 \left(p_3\pi(1,3,K-1) + p_4\pi(1,4,K-1)\right) \\
& \quad +p_2\pi(2,K) + P_A p_2 \left(p_3\pi(2,3,K-1) + p_4\pi(2,4,K-1)\right) \\
&= p_1\pi(1,K) + p_2\pi(2,K) + P_A\{p_1p_3\pi(1,3,K-1) \\
& \quad +p_1p_4\pi(1,4,K-1) + p_2p_3\pi(2,3,K-1)
\end{aligned}$$

$$+p_2p_4\pi(2,4,K-1)\} \tag{4.23}$$

By substituting Eqs.(4.20) and (4.23) into Eq.(4.19), we have

$$\begin{aligned} P_{loss}(A) &= \frac{(\alpha+\beta)}{(p_1\beta+p_2\alpha)}[p_1\pi(1,K)+p_2\pi(2,K)+P_A\{p_1p_3\pi(1,3,K-1) \\ &\quad +p_1p_4\pi(1,4,K-1)+p_2p_3\pi(2,3,K-1)+p_2p_4\pi(2,4,K-1)\}]. \end{aligned} \tag{4.24}$$

Using the same argument, we can obtain the loss probability for packets from stream B, and it is given by

$$\begin{aligned} P_{loss}(B) &= \frac{(\gamma+\delta)}{(p_3\delta+p_4\gamma)}[(p_3\pi(3,K)+p_4\pi(4,K)+P_B\{p_1p_3\pi(1,3,K-1) \\ &\quad +p_1p_4\pi(1,4,K-1)+p_2p_3\pi(2,3,K-1)+p_2p_4\pi(2,4,K-1)\}]. \end{aligned} \tag{4.25}$$

For the loss probability $P_{loss}(O)$ seen by an arbitrary arrival, we have

$$P_{loss}(O) = \lim_{t\to\infty}\frac{1}{N(t)}\int_0^t U(s)dN(s) = \lim_{t\to\infty}\frac{t}{N(t)}\lim_{t\to\infty}\frac{1}{t}\int_0^t U(s)dN(s) \tag{4.26}$$

Since $N(t) = N_A(t) + N_B(t)$, we obtain

$$\lim_{t\to\infty}\frac{N(t)}{t} = \lim_{t\to\infty}\frac{N_A(t)}{t} + \lim_{t\to\infty}\frac{N_B(t)}{t} = \frac{p_1\beta+p_2\alpha}{\alpha+\beta} + \frac{p_3\delta+p_4\gamma}{\gamma+\delta} \tag{4.27}$$

and

$$\begin{aligned} \lim_{t\to\infty}\frac{1}{t}\int_0^t U(s)dN(s) &= \lim_{t\to\infty}\frac{1}{t}\int_0^t U(s)dN_A(s) + \lim_{t\to\infty}\frac{1}{t}\int_0^t U(s)dN_B(s) \\ &= p_1\pi(1,K)+p_2\pi(2,K)+p_3\pi(3,K)+p_4\pi(4,K) \\ &\quad +p_1p_3\pi(1,3,K-1)+p_1p_4\pi(1,4,K-1) \\ &\quad +p_2p_3\pi(2,3,K-1)+p_2p_4\pi(2,4,K-1) \end{aligned} \tag{4.28}$$

By substituting Eqs.(4.27) and (4.28) into Eq.(4.26), we can easily obtain $P_{loss}(O)$.

In order to obtain the packet loss probabilities $P_{loss}(A)$, $P_{loss}(B)$ and $P_{loss}(O)$, we need to obtain the limiting probability $\pi(i,j,q)$ of the Markov process $\{Y_A(n-), Y_B(n-), Z(n-)\}$. For this purpose, we use direct numerical methods as in the continuous-time case.

The input stream A is described by a two-state Markov chain with transition probability matrix $\mathbf{B_1}$ given by

$$\mathbf{B_1} = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}. \tag{4.29}$$

The transition probability matrix for the input stream B, $\mathbf{B_2}$, is given by

$$\mathbf{B_2} = \begin{pmatrix} 1-\gamma & \gamma \\ \delta & 1-\delta \end{pmatrix}. \tag{4.30}$$

The aggregated input process is fully characterized by the product chain of these two independent two-state streams. The transition probability matrix $\mathbf{B}$ of the product chain is then given by the Kronecker product

$$\mathbf{B} = \mathbf{B_1} \otimes \mathbf{B_2} = \begin{pmatrix} (1-\alpha)(1-\gamma) & (1-\alpha)\gamma & \alpha(1-\gamma) & \alpha\gamma \\ (1-\alpha)\delta & (1-\alpha)(1-\delta) & \alpha\delta & \alpha(1-\delta) \\ \beta(1-\gamma) & \beta\gamma & (1-\beta)(1-\gamma) & (1-\beta)\gamma \\ \beta\delta & \beta(1-\delta) & (1-\beta)\delta & (1-\beta)(1-\delta) \end{pmatrix}. \tag{4.31}$$

As in the continuous-time case, we represent the system state by (i,j,q), where i is the state of stream A, j is the state of stream B, and q is the number of packets in the system. For convenience, let $\mathbf{q}$ denote the set of states for a given value of q $\{(i,j,q), i \in \{1,2\}, j \in \{3,4\}\}$. The transition probability matrix $\mathbf{P}$ of our system is then given in Figure 4.4. In Figure 4.4, P_{ia} represents the probability that i number of packets arrive, and P_{id} represents the probability that i number of packets depart. (a denotes "arrival," d denotes "departure," and i represents the number of arrivals

$$
\begin{array}{c|cccccccc|}
 & 0 & 1 & 2 & 3 & \cdots & K-2 & K-1 & K \\
0 & (P_{1a}P_{1d}+P_{0a})B & (P_{1a}P_{0d}+P_{2a}P_{1d})B & P_{2a}P_{0d}B & & & & & \\
1 & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{1a}P_{1d})B & (P_{1a}P_{0d}+P_{2a}P_{1d})B & P_{2a}P_{0d}B & & & & \\
2 & & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{1a}P_{1d})B & (P_{1a}P_{0d}+P_{2a}P_{1d})B & P_{2a}P_{0d}B & & & \\
\cdot & & \searrow & \searrow & \searrow & \searrow & & & \\
\cdot & & & & & \vdots & & & \\
\cdot & & & & & & & & \\
K-1 & & & & & & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{0a}P_{1d})B & P_{0a}P_{0d}B \\
K & & & & & & & P_{1d}B & P_{0d}B
\end{array}
$$

Figure 4.4: Transition Probability Matrix for 2-stream Case (Discrete Time)

or departures.) P'_{0a} represents the probability that some number of packets arrive. When $Y_A = i$ $(i = 1,2)$ and $Y_B = j$ $(j = 3,4)$, we have $P_{0a} = (1-p_i)(1-p_j)$, $P_{1a} = p_i(1-p_j) + p_j(1-p_i)$, $P_{2a} = p_i p_j$, and $P'_{0a} = 1 - P_{0a} = 1 - (1-p_i)(1-p_j)$. Further, $P_{0d} = 1 - s$ and $P_{1d} = s$.

By solving the set of steady state equations $\pi\mathbf{P} = \pi, \pi\mathbf{e} = 1$ where $\pi = [\pi(1,3,0), \pi(1,4,0), \pi(2,3,0), \pi(2,4,0), \ldots, \pi(2,3,K), \pi(2,4,K)]$, we can obtain the steady-state probability $\pi(i,j,q)$. Then, the marginal distribution such as $\pi(i,q) = \sum_j \pi(i,j,q)$ can easily be computed.

Note that for a discrete-time analysis, the extra complication of simultaneous arrivals is accounted for quite easily with the stochastic integral approach used in this chapter. Our analysis can easily be extended to accommodate a wide range of packet discarding schemes. In fact, any state-based packet discarding schemes can be analyzed in the same fashion. As an example, a priority packet discarding scheme [YLS88, WK90], a technique frequently proposed for high-speed networks, is analyzed in the following section.

4.3 Threshold-Based Priority Packet Discarding Scheme

In this section, our analyses presented in Sections 4.1 and 4.2 are extended to accommodate a priority packet discarding scheme. As shown in section 2.4.2, a priority packet discarding scheme is a popular congestion control technique for high-speed networks. It can be used to satisfy varying loss requirements of different classes of traffic. In general, loss-sensitive traffic such as data is given priority over loss-insensitive traffic such as voice. When network congestion occurs, varying loss requirements of different classes of traffic can be satisfied by selectively discarding low priority packets.

For voice or video traffic, a priority packet discarding scheme may be used in conjunction with an embedded coding technique mentioned in section 2.4.2. If an embedded coding technique is used for voice, the encoded information is divided into more significant bits and less significant bits. More significant bits form high priority packets, and less significant bits form low priority packets. If an embedded coding technique is used for video, low frequency components of video form high priority packets, and high frequency components of video (refinement of image) form low priority packets. With an embedded coding, packets containing more important information are given higher priority than packets containing less important information, and when network congestion occurs, packets containing less important information are discarded first.

In this section, a simple threshold-based discarding scheme [BCS90] is considered. With this scheme, low priority packets are accepted only if the current system occupancy is less than a certain threshold, θ.

Again it is assumed that two heterogeneous streams (stream A and stream B) are multiplexed. The extension of the analysis for $N(>2)$ stream case is discussed in Appendix D.

4.3.1 Continuous-Time Case

The same model and notations used in Section 4.1 are assumed in this subsection. Recall that $P_A(q)$ ($P_B(q)$) denotes the long term probability that an arrival from stream A (B) sees the system in state q. Assume that stream A has a higher priority than stream B. Then,

$$P_{loss}(A) = P_A(K) \quad \text{and} \quad P_{loss}(B) = \sum_{q=\theta}^{K} P_B(q). \tag{4.32}$$

From Eqs.(4.10), (4.11), and (4.32), we have

$$P_{loss}(A) = \frac{(\alpha+\beta)(\lambda_1\pi(1,K) + \lambda_2\pi(2,K))}{\lambda_1\beta + \lambda_2\alpha} \tag{4.33}$$

and

$$P_{loss}(B) = \frac{(\gamma+\delta)(\lambda_3\sum_{q=\theta}^{K}\pi(3,q) + \lambda_4\sum_{q=\theta}^{K}\pi(4,q))}{\lambda_3\delta + \lambda_4\gamma}. \tag{4.34}$$

For $i, i' \in \{1,2\}$, $j, j' \in \{3,4\}$, and $q, q' \in \{0,1,2,...,K\}$, the infinitesimal generator $\mathbf{Q}$ is given by

$$\mathbf{Q}_{(i,j,q)\rightarrow(i',j',q')} = \begin{cases} r_{ii'}, & \text{if } i' \neq i, j' = j, q' = q \\ r_{jj'}, & \text{if } i' = i, j' \neq j, q' = q \\ \lambda_i + \lambda_j, & \text{if } i' = i, j' = j, q' = q+1, q < \theta \\ \lambda_i, & \text{if } i' = i, j' = j, q' = q+1, q \geq \theta \\ \mu, & \text{if } i' = i, j' = j, q' = q-1 \\ -r_{i\bar{i}} - r_{j\bar{j}} - \lambda_i - \lambda_j, & \text{if } i' = i, j' = j, q' = q = 0 \\ -\mu - r_{i\bar{i}} - r_{j\bar{j}}, & \text{if } i' = i, j' = j, q' = q = K \\ -\mu - r_{i\bar{i}} - r_{j\bar{j}} - \lambda_i - \lambda_j, & \text{if } i' = i, j' = j, q' = q, 0 < q < \theta \\ -\mu - r_{i\bar{i}} - r_{j\bar{j}} - \lambda_i, & \text{if } i' = i, j' = j, q' = q, \theta < q < K \\ 0, & \text{otherwise.} \end{cases} \tag{4.35}$$

4.3.2 Discrete-Time Case

The same model and notations used in Section 4.2 are assumed in this subsection. Assume that stream A has a higher priority than stream B. For the high priority stream A,

$$U(n) = U_K(n) \tag{4.36}$$

since arriving high priority packets are discarded only when the system size becomes K. For the low priority stream B,

$$U(n) = \sum_{q=\theta}^{K} U_q(n) \tag{4.37}$$

since arriving low priority packets are always discarded when the system occupancy is greater than or equal to θ. For the remaining derivation, the same analytical technique used in subsection 4.2.2 applies. We have

$$P_{loss}(A) = \frac{(\alpha+\beta)(p_1\pi(1,K) + p_2\pi(2,K))}{p_1\beta + p_2\alpha} \tag{4.38}$$

and

$$P_{loss}(B) = \frac{(\gamma+\delta)(p_3\sum_{q=\theta}^{K}\pi(3,q) + p_4\sum_{q=\theta}^{K}\pi(4,q))}{p_3\delta + p_4\gamma}. \tag{4.39}$$

The transition probability matrix **P** for this system is given in Figure 4.5. P^h_{ia} represents the probability that i number of high priority packets arrive. When $Y_A = i$ $(i = 1, 2)$, we have $P^h_{0a} = (1 - p_i)$ and $P^h_{1a} = p_i$. For the rest of the notation used in Figure 4.5, refer to subsection 4.2.2.

Note that the method used in our analysis, a new stochastic integral approach, is easily applied to derive individual packet loss probabilities when a packet discarding control scheme is employed.

$$
\begin{array}{c|ccccccccccc|}
 & 0 & 1 & 2 & \cdots & \theta-1 & \theta & \theta+1 & \theta+2 & \cdots & K-1 & K \\
0 & (P_{1a}P_{1d}+P_{0a})B & (P_{1a}P_{0d}+P_{2a}P_{1d})B & P_{2a}P_{0d}B & & & & & & & & \\
1 & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{1a}P_{1d})B & (P_{1a}P_{0d}+P_{2a}P_{1d})B & P_{2a}P_{0d}B & & & & & & & \\
2 & & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{1a}P_{1d})B & \searrow & \searrow & & & & & & \\
\vdots & & & & \vdots & & & & & & & \\
\theta & & & & & P^{h}_{0a}P_{1d}B & (P^{h}_{0a}P_{0d}+P^{h}_{1a}P_{1d})B & P^{h}_{1a}P_{0d}B & & & & \\
\vdots & & & & \vdots & & \searrow & \searrow & \searrow & & & \\
K & & & & & & & & & & P_{1d}B & P_{0d}B
\end{array}
$$

Figure 4.5: Transition Probability Matrix for the System with Priority Packet Discarding (2-stream, Discrete Time)

4.4 Numerical Examples

4.4.1 Effects of Burstiness

In this subsection, through numerical examples, the effects of traffic characteristics on the individual packet loss probabilities are investigated. It is assumed, for simplicity, that two heterogeneous streams (stream A and stream B) are multiplexed. We show how the burstiness of one stream affects the packet loss probabilities of each of the two multiplexed streams.

As discussed in Chapter 2, burstiness is one of the most critical parameters in determining the network performance. A number of ways have been proposed to describe the burstiness of a traffic source (see section 2.2.2.1). However, consensus is yet to be reached concerning an appropriate way to describe the burstiness of a traffic source. In keeping with our focus on Markov Modulated Arrivals, we examine the following three intuitive ways to vary the burstiness of a stream. In all three we keep the mean arrival rate of the stream *constant.* The expression for the mean arrival rate of the stream is given in Eq.(4.7) for the continuous-time case and in Eq.(4.20) for the discrete-time case.

Method 1: Keep the average sojourn times in two driving states constant, and vary the arrival rates in two states. In this case, as the difference between the arrival rates in two states increases, the burstiness of the stream also increases.

Method 2: For an Interrupted Poisson Process (IPP) stream, keep the ratio of average active period (i.e., period during which packets are generated) to average idle period (i.e., period during which no packets are generated) constant, and vary both active and idle periods. In this case, as the average active and idle period increase, the burstiness of the stream also increases.

Method 3: For an IPP stream, keep the sum of the average active and idle periods constant, and vary the average active and idle periods. Since we keep the mean arrival rate constant, a smaller active period means a greater arrival rate during an active period. In this case, as the average active period decreases (i.e., as the arrival rate during an active period increases), the burstiness of the stream increases.

Note that the first and the third methods are two ways to vary peak-to-mean ratio. The peak-to-mean ratio is the most commonly used definition of the burstiness. The second method varies the average active period. The average active period is also a widely used parameter to measure the degree of the burstiness (see, for example [GRF89, HW89]).

In the following numerical examples, in order to characterize the effects of mixing bursty streams with non-bursty streams, we introduce the concept of "self-loss" for a single stream. A self-loss is the packet loss incurred when a stream is multiplexed with itself. In the figures, the following notation is used to represent the loss probabilities:

- $P_{loss}(A)$: the packet loss probability of stream A when it is multiplexed with stream B,
- $P_{loss}(B)$: the packet loss probability of stream B when it is multiplexed with stream A,
- $P_{loss}(O)$: the overall packet loss probability (i.e., the loss probability for all streams),
- self-loss(A): the packet loss probability of stream A when it is multiplexed with itself, and
- self-loss(B): the packet loss probability of stream B when it is multiplexed with itself.

In the following subsection 4.4.1.1, the results for discrete-time case are presented. (Similar results are obtained for the continuous-time case.) Results obtained in subsections 4.4.1.1 are then summarized in subsection 4.4.1.2.

4.4.1.1 Discrete-Time Case

The same model assumed in Section 4.2 is used. Let m_A denote the mean arrival rate of stream A and m_B denote the mean arrival rate of stream B.

From Eq.(4.20), we have

$$m_A = \frac{p_1\beta + p_2\alpha}{\alpha + \beta}. \tag{4.40}$$

Similarly, we have

$$m_B = \frac{p_3\delta + p_4\gamma}{\gamma + \delta}. \tag{4.41}$$

The offered load, ρ, is given by

$$\rho = \frac{m_A + m_B}{s}. \tag{4.42}$$

Throughout the numerical examples in this subsection, we assume the maximum system size $K = 10$ (i.e., buffer size = 9), the service rate $s = 0.8$, and the offered load $\rho = 0.1$. We further assume that the mean arrival rates of two streams are the same. This allows us to investigate solely the effect of burstiness. Since $m_A = m_B$, from Eq.(4.42) and $\rho = 0.1$, m_A and m_B become 0.04. In all the figures presented in this subsection, stream A is fixed, and we vary the burstiness of stream B keeping the mean arrival rate constant. A random packet discarding scheme is used.

In Figures 4.6 through 4.8, stream A is assumed to follow a geometric arrival process (i.e., $p_1 = p_2 = 0.04$). In Figure 4.6, the burstiness of stream B is varied

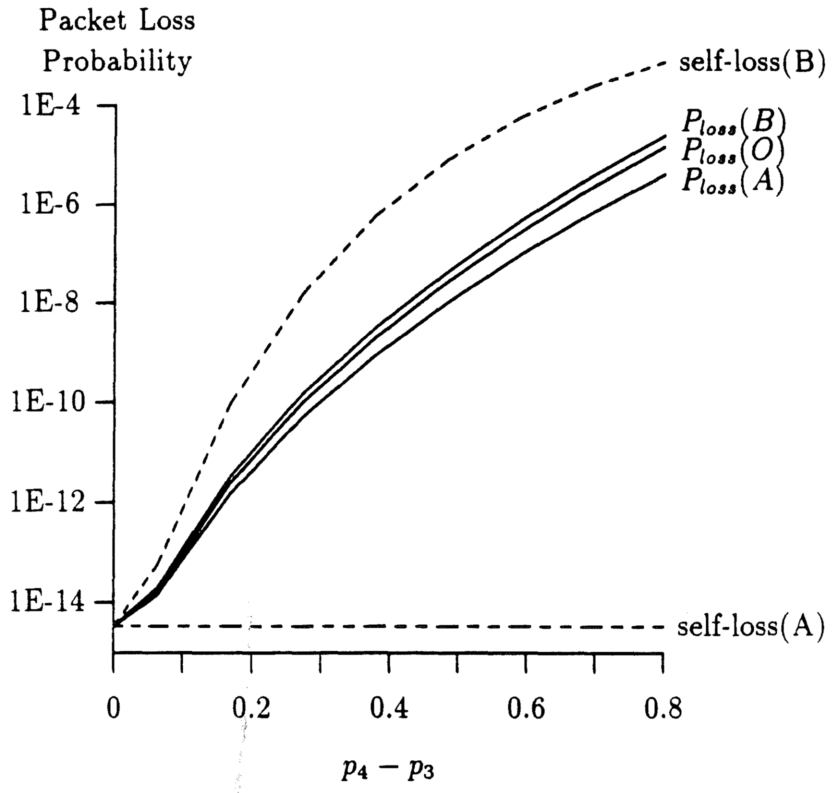


Figure 4.6: Effect of Burstiness on Packet Loss Probabilities (Discrete Time, The First Method)

using the first method described earlier. In other words, p_3 and p_4 (arrival rates in two states) are varied, keeping $\frac{1}{\gamma}$ and $\frac{1}{\delta}$ (the average sojourn times in two states) constant. The bigger the difference between p_3 and p_4, the greater the burstiness of stream B. In this figure, the values of γ and δ are 0.01 and 0.19, respectively. The horizontal axis shows the difference between p_4 and p_3. At the leftmost starting point (i.e., when $p_4 - p_3 = 0$), stream B becomes a geometric stream. In this figure, as moving to the right, the difference between p_4 and p_3 becomes larger, and thus, the burstiness of stream B increases.

Several observations can be made from Figure 4.6. At the leftmost starting point, the packet loss probabilities for stream A and stream B are the same since, at this point, stream B is also a geometric stream (i.e., $p_3 = p_4 = 0.04$). As the burstiness of stream B increases, both the loss probability of stream B and the loss probability of stream A increase. From this, it can be concluded that an increase in the burstiness of one stream negatively affects the stream itself and also the other multiplexed stream.

Next, compare the $P_{loss}(A)$ curve with the self-loss(A) curve in Figure 4.6. The self-loss(A) (i.e., the loss probability of stream A when it is multiplexed with an identical stream) is always smaller than the $P_{loss}(A)$ (i.e., the loss probability of stream A when it is multiplexed with stream B). In other words, a geometric stream (stream A) is penalized by sharing a buffer with a bursty stream (stream B), as opposed to sharing a buffer with another Poisson stream. This is because bursty stream causes buffer buildups, blocking the geometric stream.

Compare the $P_{loss}(B)$ curve with the self-loss(B) curve. The $P_{loss}(B)$ is always smaller than the self-loss(B). This shows that a bursty stream (stream B) gains (i.e., $P_{loss}(B) <$ self-loss(B)) by sharing a buffer with a geometric stream (stream A), as opposed to sharing a buffer with another bursty stream. This is because a geometric stream does not cause as much buffer buildup as a bursty stream does, and thus, a stream loses less packets when it is multiplexed with a geometric stream than when it is multiplexed with a bursty stream. From this, it can be concluded that the traffic mix has a significant effect on the packet loss probabilities.

Figure 4.6 also shows that when two different traffic streams are multiplexed, the stream with the smaller self-loss probability is penalized. In this case, self-loss(A) is smaller than self-loss(B), and stream A is penalized (i.e., $P_{loss}(A) >$ self-loss(A)). Furthermore, the bigger the difference between two self-loss probabilities, the greater the penalty.

In Figure 4.7, a discrete-time version of IPP is used for stream B (i.e., $p_3 = 0$). p_4 is equal to 0.8. The burstiness of stream B is varied using the second method described earlier. In other words, we vary $\frac{1}{\gamma}$ (the average idle period of stream B) and $\frac{1}{\delta}$ (the average active period of stream B), keeping their ratio $\frac{\delta}{\gamma}$ constant. In this case, the longer the average active (or idle) length, the greater the burstiness. In this figure, the value of $\frac{\delta}{\gamma}$ is equal to 19. The horizontal axis shows the average idle period. In this figure, as moving to the right, both average active period and idle period increase, and thus, the burstiness of stream B increases. Similar observations to those in Figure 4.6 are made in this figure. An increase in the burstiness of one stream negatively affects both the stream itself and the other stream multiplexed. When a bursty stream and a geometric stream are multiplexed together, the geometric stream is penalized, and the bursty stream benefits. The stream with the smaller self-loss probability is penalized, and the bigger the difference between two self-loss probabilities, the greater the penalty.

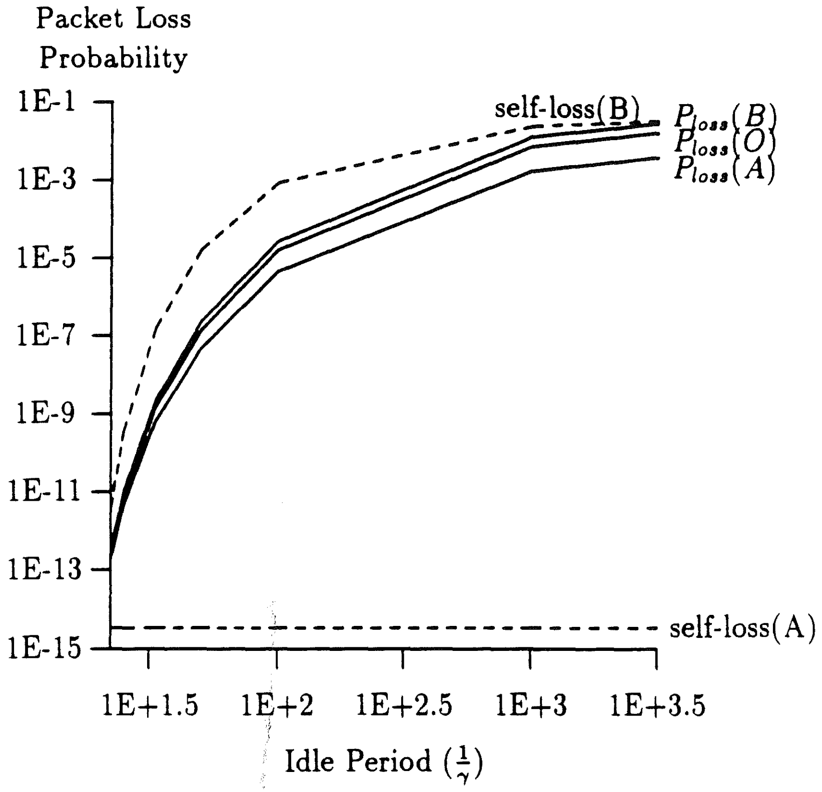


Figure 4.7: Effect of Burstiness on Packet Loss Probabilities (Discrete Time, The Second Method)

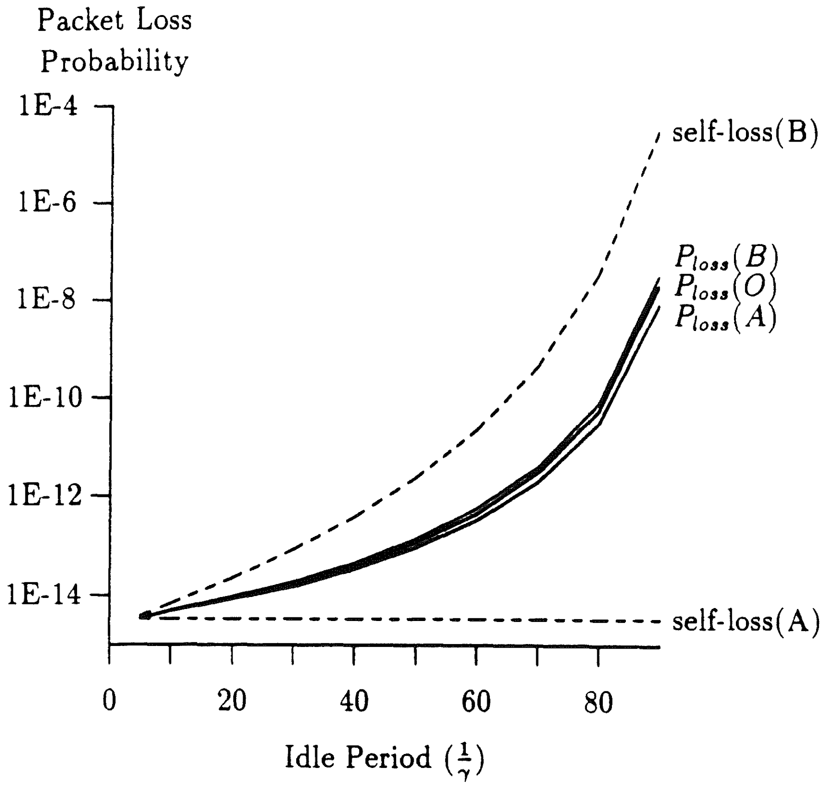


Figure 4.8: Effect of Burstiness on Packet Loss Probabilities (Discrete Time, The Third Method)

In Figure 4.8, a discrete-time version of IPP is used for stream B (i.e., $p_3 = 0$). The burstiness of stream B is varied using the third method described earlier. In other words, we vary $\frac{1}{\delta}$ (the average active period) and $\frac{1}{\gamma}$ (the average idle period), keeping $\frac{1}{\gamma} + \frac{1}{\delta}$ (the sum of average active and idle period) constant. In this figure, $\frac{1}{\gamma} + \frac{1}{\delta} = 100$. In this figure, as moving to the right, the active period ($= 1/\delta$) decreases and p_4 increases, therefore, the burstiness of stream B increases. Again, similar observations made for Figures 4.6 and 4.7 can be made for this figure.

In Figures 4.9 through 4.11, stream A is also assumed to be a non-geometric, bursty stream. Again, stream A is fixed, and the burstiness of stream B is varied keeping the mean arrival rate constant. In Figure 4.9, as in Figure 4.6, the burstiness of stream B is varied using the first method described earlier. The values of γ and δ

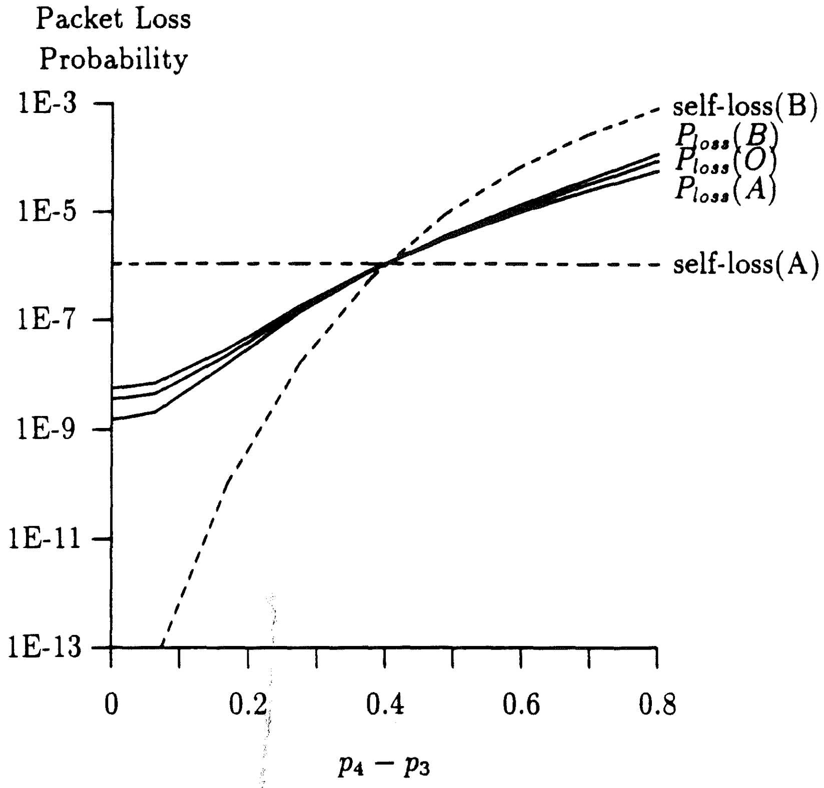


Figure 4.9: Effect of Burstiness on Packet Loss Probabilities (Discrete Time, The First Method)

are 0.01 and 0.19, respectively. For stream A, we use $\alpha = 0.01$, $\beta = 0.19$, $p_1 = 0.02$ and $p_2 = 0.42$ (i.e., $p_2 - p_1 = 0.4$). Again, an increase in the burstiness of one stream negatively affects both the stream itself and the other stream multiplexed. When $p_4 - p_3 = 0.4$, the packet loss probabilities for stream A and stream B are the same since both streams become exactly the same stream. When $p_4 - p_3 < 0.4$, stream A is more bursty than stream B, and self-loss(A) is larger than self-loss(B). In this region, stream B whose self-loss probability is smaller than that of stream A is penalized, and the bigger the difference between two self-loss probabilities, the greater the penalty. When $p_4 - p_3 > 0.4$, stream B is more bursty than stream A, and self-loss(B) is larger than self-loss(A). In this region, stream A has smaller self-loss probability, and thus, it is penalized. Again, the bigger the difference between two self-loss probabilities, the greater the penalty.

In Figure 4.10, as in Figure 4.7, the burstiness of stream B is varied using the second method described earlier. As in Figure 4.7, we use $p_3 = 0$, $p_4 = 0.8$ and $\frac{\delta}{\gamma} = 19$ for stream B. For steram A, we use $\alpha = 0.01$, $\beta = 0.19$, $p_1 = 0$, $p_2 = 0.8$, and thus, the average idle period of stream A, $\frac{1}{\alpha}$, is 100. When the average idle period of stream B, $\frac{1}{\gamma}$, is 100, the packet loss probabilities for stream A and stream B are the same since both streams become exactly the same stream. When $\frac{1}{\gamma} < 100$, stream A is more bursty than stream B, and self-loss(A) is larger than self-loss(B). When $\frac{1}{\gamma} > 100$, stream B is more bursty than stream A, and self-loss(B) is larger than self-loss(A). In both regions, the stream with the smaller self-loss probability is penalized, and the bigger the difference between two self-loss probabilities, the greater the penalty.

In Figure 4.11, as in Figure 4.8, the burstiness of stream B is varied using the third method described earlier. As in Figure 4.8, we use $p_3 = 0$ and $\frac{1}{\gamma} + \frac{1}{\delta} = 100$ for stream B. For stream A, we use $\alpha = \beta = 0.02$, $p_1 = 0$, $p_2 = 0.08$, and thus, both average idle period and average active period of stream A, $\frac{1}{\alpha}$ and $\frac{1}{\beta}$ respectively, are

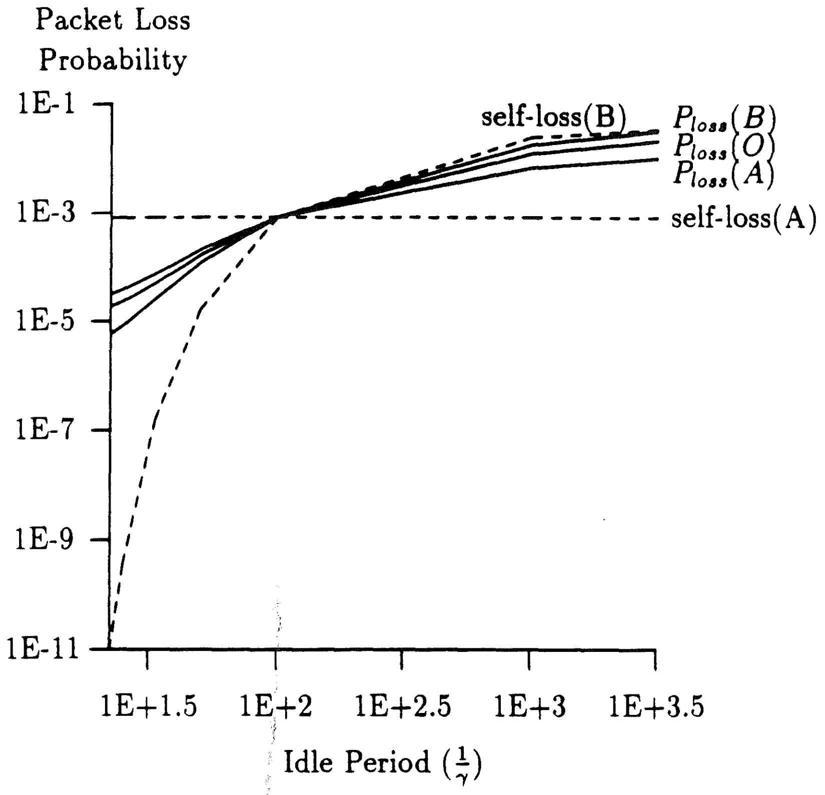


Figure 4.10: Effect of Burstiness on Packet Loss Probabilities (Discrete Time, The Second Method)

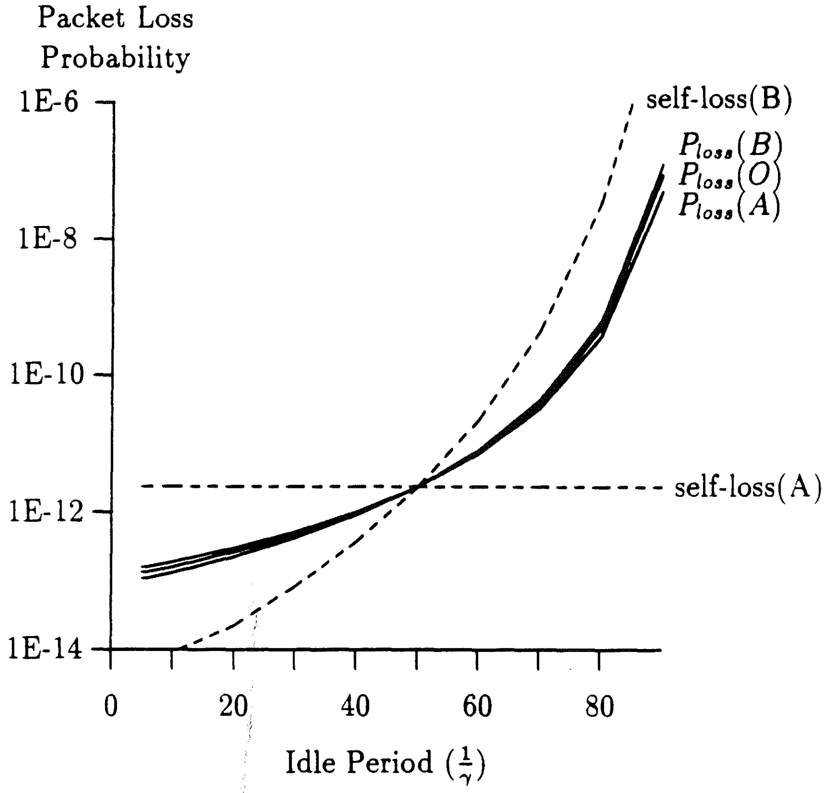


Figure 4.11: Effect of Burstiness on Packet Loss Probabilities (Discrete Time, The Third Method)

50. When the average idle period of stream B, $\frac{1}{\gamma}$, is 50, the packet loss probabilities for both streams are the same since both become exactly the same stream. Again, similar observations to those in Figures 4.9 and 4.10 can be made in this figure.

4.4.1.2 Summary

In the above subsection 4.4.1.1, the effects of traffic characteristics and the traffic mix on the packet loss probability of each of the input streams were investigated. The following summarizes the results:

- An increase in the burstiness of one stream results in an increase in the packet loss probabilities of that stream and of others which are multiplexed together.
- When two different traffic streams are multiplexed, the less bursty stream is always penalized, and the more bursty stream always benefits.
- When two different traffic streams are multiplexed together, the stream with the smaller self-loss probability is penalized. The bigger the difference between two self-loss probabilities, the greater the penalty.

Finally, note that the differences between individual loss probabilities are significant in all the figures presented in numerical example section (Figures 4.6 through 4.11). In our numerical examples, we assumed that the mean arrival rates of streams A and B are the same. However, the difference between the packet loss probabilities for stream A and B is often an order of magnitude or greater. This shows that the overall packet loss probability may not provide sufficient insight when heterogeneous traffic sources are multiplexed.

The importance of individual loss probabilities is better explained through an example. Consider admission control discussed in section 2.2.2.1 Admission control decides whether to accept or reject a new call based on whether the required performance can be maintained. If the overall packet loss probability is used as a criterion in admission control when heterogeneous traffic sources are multiplexed, the GOS of the new call may not be guaranteed. This is because, depending on the burstiness of a new coming call, its packet probability may be significantly larger than the overall packet loss probability. For example, in Figure 4.6, the difference between the packet loss probability of stream A and the overall packet loss probability is about an order of magnitude when the difference between the burstiness of two streams are the largest.

4.4.2 Effectiveness of Priority Packet Discarding

In this subsection, through numerical examples, the effectiveness of a priority packet discarding scheme is investigated. Again, it is assumed that two heterogeneous streams (stream A and stream B) are multiplexed. Only the results for discrete-time case are presented here. (Similar results are obtained for the continuous-time case.)

Throughout the numerical examples in this subsection, we assume $K = 20, s = 0.8$ and $\rho = 0.75$, unless otherwise stated. We further assume that the mean arrival rate of stream A, m_A, and the mean arrival rate of stream B, m_B, are the same. $m_A = m_B = 0.3$ since the offered load $\rho = 0.75$. Stream A is assumed to follow a geometric arrival process (i.e., $p_1 = p_2 = 0.3$) and stream B is a bursty stream. For stream B, we assume $\gamma = \delta = 0.01$, $p_3 = 0.2$ and $p_4 = 0.4$, unless otherwise stated. Priority is given to the bursty stream B, and thus, stream A packets are discarded when the system size is greater than or equal to a given threshold value θ.

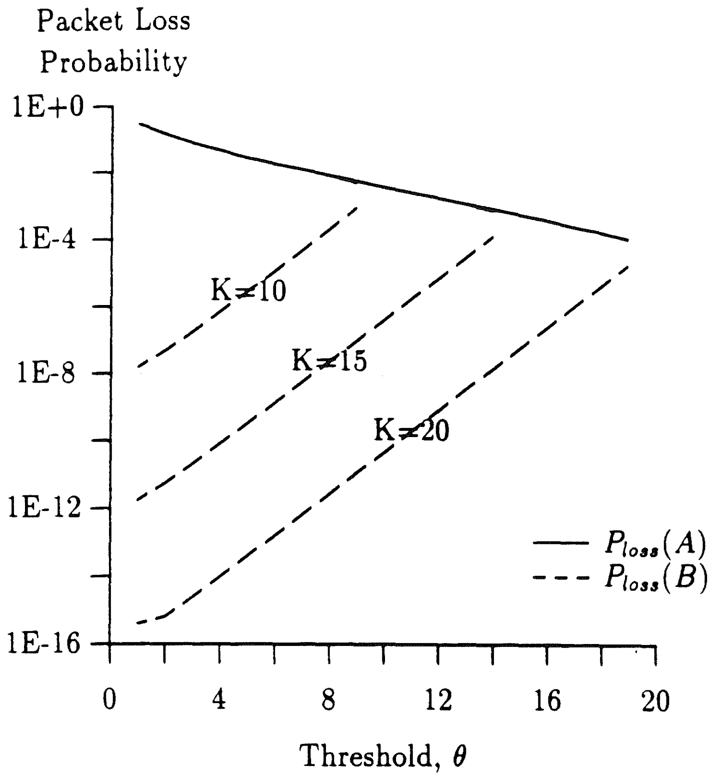


Figure 4.12: Packet Loss Probabilities With Priority Packet Discarding

Figure 4.12 shows the individual loss probabilities for various values of K. The horizontal axis represents θ values. For instance, if the GOS requirement of the geometric stream (stream A) is 10^{-3} packet loss probability and that of the bursty stream (stream B) is 10^{-9}, the system size of 20 can satisfy both requirements by setting the threshold value θ to 12. Compare this figure with Figure 4.13, which shows the packet loss probabilities in terms of the maximum system size when no priority packet discarding is employed. Without priority packet discarding, to meet the same GOS requirements, the required system size is increased to 48. This illustrates the effectiveness of priority packet discarding scheme.

Figure 4.14 also shows the effectiveness of priority packet discarding scheme. In this figure, the bursty stream B is fixed, and the mean arrival rate of the geometric

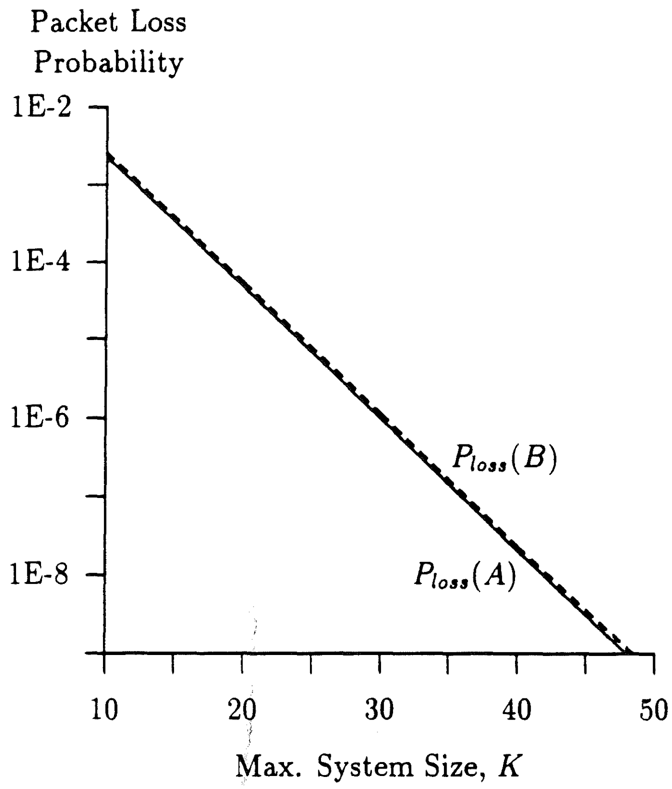


Figure 4.13: Packet Loss Probabilities vs. Max. System Size Without Priority Packet Discarding

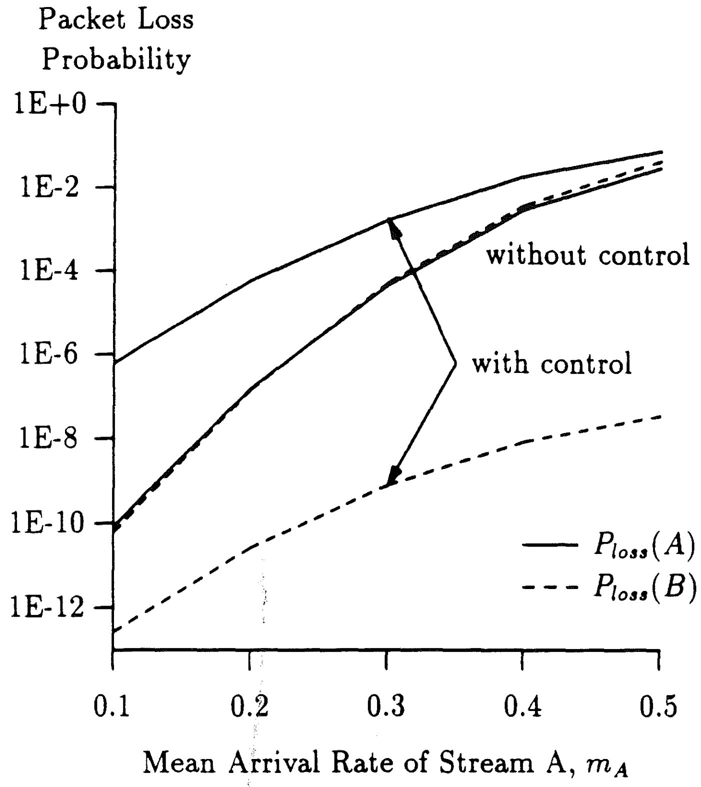


Figure 4.14: Effectiveness of Priority Packet Discarding

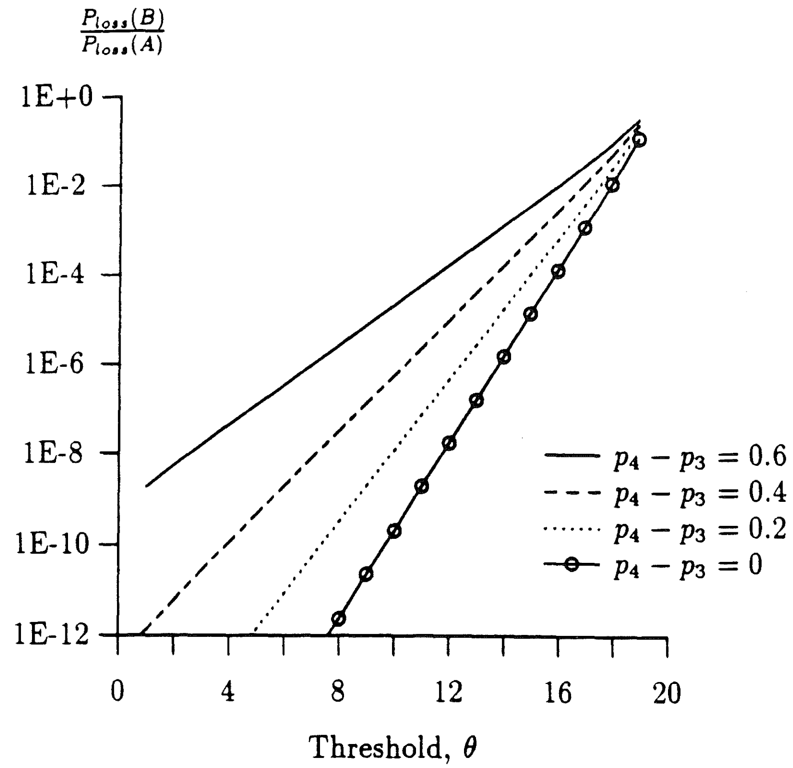


Figure 4.15: Effect of Burstiness of High-Priority Stream on Priority Packet Discarding

stream A is varied (i.e., p_1 and p_2 are varied). The threshold value θ is set to 12. Again, let us assume that the GOS requirement of the geometric stream A is 10^{-3} packet loss probability and that of the bursty stream B is 10^{-9}. With priority packet discarding, mean arrival rate of stream A up to 0.3 is acceptable, whereas without priority packet discarding, acceptable mean arrival rate of stream A is less than 0.13. From Figures 4.12 through 4.14, it can be concluded that with priority packet discarding, either the required buffer space is decreased or the acceptable offered load of low priority stream is increased.

Figure 4.15 illustrates the effect of burstiness of high priority stream on the effectiveness of priority packet discarding. In this figure, the geometric stream A is fixed, and the burstiness of stream B is varied, keeping the mean arrival rate constant. ($m_A = m_B = 0.3$.) The burstiness of stream B is varied using the first method described in subsection 4.4.1; p_3 and p_4 (arrival rates in two states) are varied, keeping $\frac{1}{\gamma}$ and $\frac{1}{\delta}$ (the average sojourn times in two states) constant. The bigger the difference between p_3 and p_4, the greater the burstiness of stream B. The vertical axis shows $\frac{p_{loss}(B)}{p_{loss}(A)}$, the ratio of stream B packet loss probability to stream A packet loss probability. For a given threshold value, a smaller ratio means a greater difference between high-priority and low-priority packet loss probabilities and thus implies greater effectiveness of the priority packet discarding. From this figure, it can be seen that for a given threshold value, the more bursty stream B is, the greater this ratio becomes. Furthermore, if stream B is more bursty, the threshold value for stream A needs to be decreased in a lager amount to obtain the same order of degree decrease in this ratio. For instance, to decrease this ratio by two orders of magnitude, the most bursty stream in this figure ($p_4 - p_3 = 0.6$) requires threshold value to be decreased by 4 whereas the least bursty stream ($p_4 - p_3 = 0$) requires threshold value to be decreased by 2. Therefore, it can be concluded from this figure that if high priority stream is very bursty, selective discarding of low priority packets has low effectiveness, i.e., the packet loss probability of high priority stream does not decrease much.

Figure 4.16 illustrates the effect of burstiness of low priority stream on the effectiveness of priority packet discarding. In this figure, stream A is also assumed to be a bursty stream. The high priority stream (Stream B) is fixed, and the burstiness of the low priority stream (stream A) is varied, keeping the mean arrival rate constant. ($m_A = m_B = 0.3$.) The burstiness of stream A is varied using the first method described in subsection 4.4.1. From this figure, it is observed that a change in the burstiness of low priority stream causes only a slight change in the ratio of packet

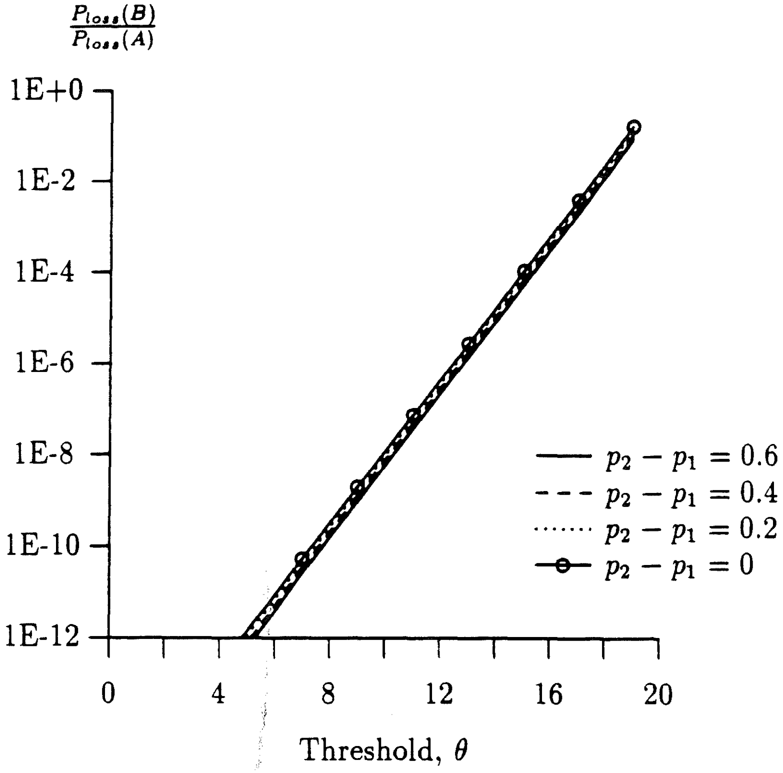


Figure 4.16: Effect of Burstiness of Low-Priority Stream on Priority Packet Discarding

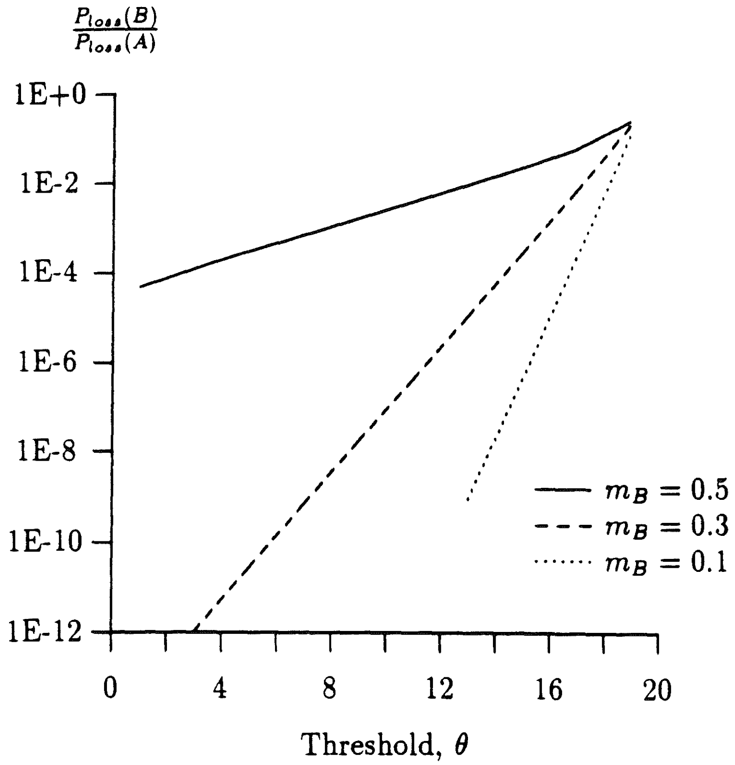


Figure 4.17: Effect of High-Priority Load on Priority Packet Discarding

loss probabilities. From Figures 4.15 and 4.16, it can be concluded that the the effectiveness of priority packet discarding highly depends on the burstiness of high priority stream, but it does not depend much on the burstiness of low priority stream.

Figure 4.17 shows the effect of high priority load on the effectiveness of priority packet discarding. In this figure, the mean arrival rate of the low priority stream (stream A) is fixed to 0.3, and the mean arrival rate of the high priority stream (stream B) is varied. We assume $\gamma = \delta = 0.01$ and $\frac{p_4}{p_3} = 3$. Again, the vertical axis shows the ratio of packet loss probabilities, $\frac{p_{loss}(B)}{p_{loss}(A)}$. From this figure, it can be seen that this packet loss probability ratio is very sensitive to the offered load of high priority stream; as the offered load of high priority stream increases, the priority packet discarding scheme becomes less effective.

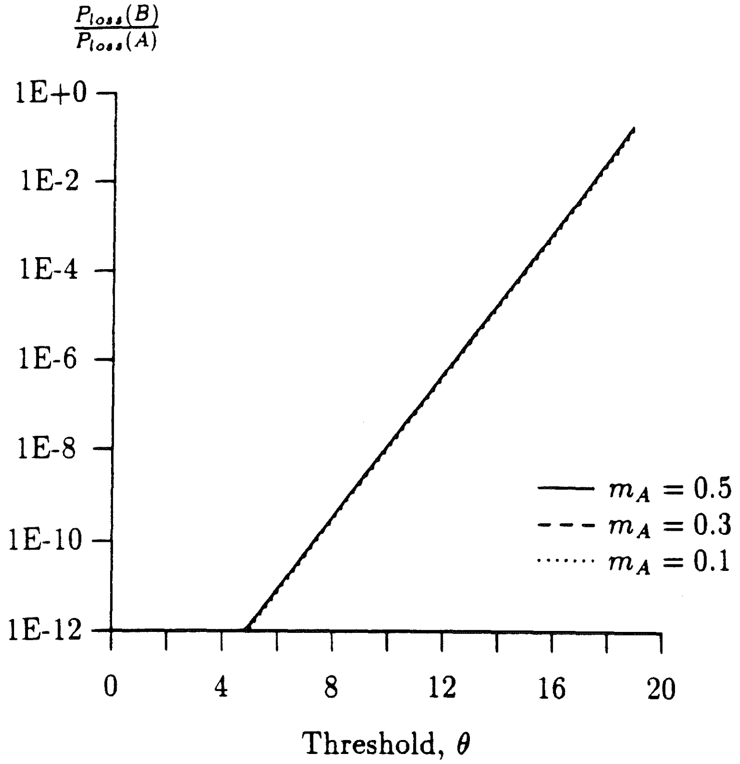


Figure 4.18: Effect of Low-Priority Load on Priority Packet Discarding

Figure 4.18 shows the effect of low priority load on the effectiveness of priority packet discarding. In this figure, the mean arrival rate of the high priority stream (stream B) is fixed to 0.3, and the mean arrival rate of the low priority stream (stream A) is varied. For stream B, we assume $\gamma = \delta = 0.01$, $p_3 = 0.2$ and $p_4 = 0.4$. From this figure, it is observed that the packet loss probability ratio is not affected by the offered load of low priority stream. From Figures 4.17 and 4.18, it can be concluded that the effectiveness of priority packet discarding is affected by the offered load of high priority stream, but not by the offered load of low priority stream.

Figure 4.19 shows the maximum offered load as a function of GOS requirement of the high priority stream (stream B). In this figure, the same geometric stream is used for both stream A and stream B (i.e., $p_1 = p_2 = p_3 = p_4$). The GOS requirement

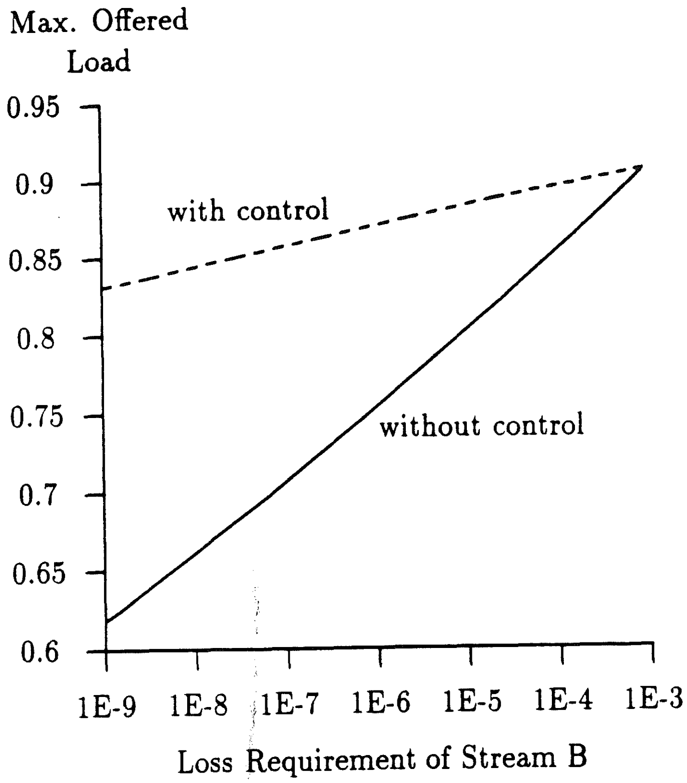


Figure 4.19: Max. Offered Load vs. Loss Requirement of High-Priority Stream

of stream A is assumed to be 10^{-3} packet loss probability, and the GOS requirement of stream B is varied. When no priority packet discarding is employed, the most stringent packet loss requirement needs to be satisfied for all the packets. For the system with priority packet discarding, a threshold value which gives the maximum offered load is used. From this figure, it can be seen that the performance gain increases as the difference between the GOS requirements of stream A and stream B increases. From Figures 4.15 through 4.19, it can be concluded that the priority packet discarding becomes most effective when the high priority stream is not bursty, the offered load of high priority stream is the smallest, and the difference between the GOS requirements is the greatest.

4.5 Chapter Summary

In this chapter, we considered a finite buffer queueing system with heterogeneous arrival streams to investigate how the burstiness affects the packet loss in individual streams. We examined the class of MMA streams both in continuous time (a MMPP) and in discrete time (a MMBP) and presented an exact analysis for the packet loss for each MMA stream. Our method of analysis used a new stochastic integral approach and was general enough to allow a similar treatment of both continuous and discrete-time cases. Our analysis method was also applied to a priority packet discarding scheme, a congestion control mechanism suitable for high-speed networks.

The concept of self-loss, the packet loss incurred when a stream is multiplexed with an identical stream, was introduced to study the effects of the traffic burstiness on the packet loss. The results showed that an increase in the burstiness of one stream negatively affects both the stream itself and the other streams which are multiplexed together. It was also shown that when two different traffic streams are multiplexed, the less bursty stream is always penalized. Furthermore, the bigger the difference between the self-loss probabilities of the two streams, the greater the penalty.

The effectiveness of a priority packet discarding scheme was also investigated. The research showed that with priority packet discarding, more traffic can be supported without violating a given service requirement. It was also shown that the priority packet discarding scheme becomes most effective when the high priority stream is not bursty, the offered load of the high priority stream is small, and the difference between the GOS requirements of high and low priority streams is large. The traffic characteristics of the low priority stream, such as the burstiness and the mean arrival rate, have almost no (or little) effect on the effectiveness of priority packet discarding.

Chapter 5

Conclusions and Future Research Problems

In high-speed networks, the ratio of processing time to cell transmission time and the ratio of propagation delay to cell transmission time are increased. Due to these changes, new network architectures are required for high-speed networks. In this dissertation, we focused on two factors which significantly affect the design of high-speed networks: error recovery schemes and congestion control schemes. In the following two sections, we summarize our research contributions and discuss future research problems.

5.1 Summary

We first investigated the effects of protocol-processing overhead on the performance of error recovery schemes in high-speed network environments (Chapter 3). Two error recovery schemes have been studied: an edge-to-edge scheme and a link-by-link scheme. In the link-by-link scheme, error recovery is performed at each switching node, and thus, retransmissions of an erred packet take place between two adjacent switching nodes. The edge-to-edge scheme moves the error recovery function to a higher end-to-end protocol layer, reducing the protocol processing overhead at switching nodes within a network. We investigated these two error recovery schemes and established the effectiveness of edge-to-edge scheme for high-speed networks (i.e., networks with high-speed/low-error-rate channels).

We also developed an analytical framework for a priority packet discarding scheme, a traffic control scheme suitable for high-speed networks (Chapter 4). Our framework explicitly models different service classes and bursty traffic sources. The loss probability of each arrival stream is obtained for both continuous-time and discrete-time cases. A new characterization of an arrival stream, which we refer to as self-loss, is introduced to qualitatively predict the effects of multiplexing bursty streams with non-bursty streams. In this research, the impact of traffic heterogeneity on the performance of individual traffic stream is quantified, and the effectiveness of a priority packet discarding scheme is established.

In summary, in this dissertation, we attacked key research issues in high-speed networks and provided significant insights into optimal design of high-speed networks.

5.2 Future Work

There are some possible extensions to our work presented in this dissertation. In the following, we list some of the possible extensions:

- In the analytical model used in Chapter 3, we assumed that the generation of new packets follows a Poisson process. A Poisson process may be somewhat restrictive to represent multimedia traffic. The next step of our research is to incorporate multimedia traffic into the analytical model.

- The present model in Chapter 4 analyzed one node and obtained individual packet loss probabilities. Possible future work would consider a network-of-queues and obtain individual packet loss probabilities on the end-to-end basis.
- The analysis presented in Chapter 4 relies on solving a set of linear equations. This calculation needs to be done fairly quickly when the priority schemes are implemented on a real time basis. It would require development of efficient computation techniques.
- In Chapter 4, we introduced priorities to differentiate various types of services. When a large portion of traffic consists of the high priority traffic, however, the performance for the low priority traffic may be severely degraded. In such a case, a segregation scheme (i.e., partitioning network resources according to the given quality class) may provide desirable GOS more efficiently than the priority scheme. The performance of the priority scheme needs to be compared against to that of the segregation scheme.

Finally, it should be noted that there are still a number of unsolved research problems in the area of high-speed networks such as ATM networks. The following presents some of the future research problems:

- Local congestion control schemes need to be analyzed with the considerations that most bursty sources will likely be rate controlled.
- In most of the past work, congestion is measured through a queue length. In other words, if a queue length exceeds some threshold value, the network is considered congested. However, the queue length may not reflect the status of the network accurately in an ATM network, where the bursty traffic may cause rapid and dynamic fluctuations in the buffer occupancy. A queue length may lead to false detection of network congestion. Parameters which allow more accurate detection of congestion need to be investigated.

- Better traffic modeling for bursty and correlated arrivals is important. Experiments/measurements of multimedia system are needed to provide more accurate and realistic traffic models.

- In most of the past analytic work, performance of a single switching node is investigated. The end-to-end performance analysis of a more general network topology needs to be investigated. This may require the analysis of an output process of a multiplexer with non-renewal input processes.

- Traffic control in internetwork environment is an open question. When a high-speed network such as an ATM network interconnects existing local area networks such as Ethernet and token ring, traffic control at bridges/gateways is a challenge because of the channel bandwidth mismatch and network architectural differences between the high-speed network and interconnected LANs.

Bibliography

[ABBGRTY86] M. Accetta, R. Baron, W. Bolosky, D. Golub, R. Rashid, A. Tevanian, and M. Young, "Mach: A New Kernel Foundation for UNIX Development," Proc. of the Summer USENIX Conference, July 1986.

[AKHT87] S. Akhtar, "Congestion Control in a Fast Packet Switching Network," Master's Thesis, Washington University, December 1987.

[AKT89] H. Ahmadi, P. Kermani and P. Tran-gia, "Throughput Analysis of a Class of Selective Repeat Protocols in High-Speed Environments," Proc. IEEE GLOBECOM '89, pp.26.4.1-26.4.9.

[ALLE78] A. O. Allen, *Probability, Statistics and Queueing Theory*, New York, Academic Press, 1978.

[AMS82] D. Anick, D. Mitra and M. M. Sondhi, "Stochastic Theory of a Data-handling System with Multiple Source," Bell Syst. Tech. J., pp.1871-1894, October 1982.

[BC89] R. Ballart and Y-C. Ching, "SONET: Now It's the Standard Optical Network," IEEE Commun. Magazine, pp.8-15, March 1989.

[BCS90] K. Bala, I. Cidon and K. Shoraby, "Congestion Control for High Speed Packet Switched Network," Proc. IEEE INFOCOM '90, pp.520-526.

[BE89] P. Bhattacharya and A. Ephremides, "Optimal Scheduling with Strict Deadlines," IEEE Trans. on Automatic Control, Vol.34, No.7, pp.721 -728, July 1989.

[BKTV88] A. Bhargava, J. F. Kurose, D. Towsley, and G. Vanleemput, "Performance Comparison of Error Control Schemes in High-Speed Computer Communication Networks," IEEE J. Select. Area. Commun., Vol.6, No.9, pp.1565-1575, December 1988.

[BLM87] M. W. Beckner, T. T. Lee and S. E. Minzer, "A Protocol and Prototype for Broadband subscriber access to ISDNs," Proc. ISS '87, pp.B6.3.1-B6.3.8.

[BRAD88] P. T. Brady, "Performance of an Edge-to-Edge Protocol in a Simulated X.25/X.75 Packet Network," IEEE J. on Select. Areas in Commun., Vol.SAC-6, No.1, January 1988.

[BREM81] P. Bremaud, *Point Processes and Queues: Martingale Dynamics*, Springer-Verlag, New York, 1981.

[BS90a] T. Bradley and T. Suda, "Performance of Error Recovery Schemes in a Fast Packet Switching Network: Simulation Study," Proc. IEEE ICC '90, pp.308.2.1-308.2.6.

[BS90b] T. Bradley and T. Suda, "Survey of Unified Approaches to Integrated-Service Networks," Proc. IEEE ITS '90, pp.15.1.1-15.1.7.

[CCITT88a] CCITT, "Draft recommendation I.121-Broadband aspects of ISDN," CCITT, TD49 (PLEN), Seoul, Korea, February 1988.

[CCITT88b] CCITT Recommendations, G707, G708 and G709, 1988.

[CCITT89a] CCITT Study Group XVIII, "Meeting Report of Sub-Working Party 8/1 ATM," Temporary Document No.14-E (Plenary), Geneva, Switzerland, June 1989.

[CCITT89b] CCITT Study Group XVIII, "WP XVIII/8—Report of Meeting," Temporary Document No.16 (Plenary), Geneva, Switzerland, June 1989.

[CCITT89c] CCITT Study Group XVIII, "Report of SWP-8/4 Network Aspects," Temporary Document No.18 (Plenary), Geneva, Switzerland, June 1989.

[CCITT90] CCITT Study Group XVIII, Draft Recommendation, May 1990.

[CG88] I. Cidon and I. S. Gopal, "PARIS: An Approach to Integrated High-Speed Private Networks," Int. J. Digital & Analog Cabled Systems, Vol.1, No.2, pp.77-86, April-June, 1988.

[CGGK90] I. Cidon, I. Gopal, M. Gopal and S. Kutten, "Distributed Control in PARIS," Proc. 9th Annual Symposium on Principles of Distributed Computing, Quebec, Canada, 1990.

[CHOI89] T. Y. Choi, "Statistical Multiplexing of Bursty Sources in an ATM Network," Multimedia '89.

[CINL75] E. Cinlar, *Introduction to Stochastic Processes*, Prentice Hall, 1975.

[CJRS89] D. D. Clark, V. Jacobson, J. Romkey, and H. Salwen, "An Analysis of TCP Processing Overhead," IEEE Communications Magazine, June 1989.

[CKT89] R. Chipalkatti, J. F. Kurose and D. Towsley, "Scheduling Policies for Real-Time and Non-Real-Time Traffic in a Statistical Multiplexer," Proc. IEEE INFOCOM '89, pp.774-783.

[COME91] D. E. Comer, *Internetworking with TCP/IP Vol II: Design, Implementation and Internals*, Prentice Hall, 1991.

[CWM89] T. M. Chen, J. Walrand and D. G. Messerschmitt, "Dynamic Priority Protocols for Packet Voice," IEEE J. Select. Areas Commun., Vol.7, No.5, June 1989.

[DJ88] L. Dittmann and S. B. Jacobsen, "Statistical Multiplexing of Identical Bursty Sources in an ATM Network," Proc. IEEE GLOBECOM '88, pp.39.6.1-39.6.5.

[DL86] J. N. Daigle and J. D. Langford, "Models for Analysis of Packet Voice Communications Systems," IEEE J. Select. Areas Commun., Vol.SAC-4, No.6, pp.847-855, September 1986.

[DS89] S. Dravida and K. Sriram, "End-to-End Performance Models for Variable Bit Rate Voice over Tandem Links in Packet Networks," Proc. IEEE INFOCOM '89, pp.1089-1095.

[ECKB83] A. E. Eckberg, "Generalized Peakedness of Teletraffic Processes," Proc. 10th Intl. Teletraffic Congress, Montreal, 1983.

[EGL88] B. Eklundh, I. Gard and G. Leijonhufvud, "A Layered Architecture for ATM Networks," Proc. IEEE GLOBECOM '88, pp.13.1.1-13.1.6.

[EKIK89] H. Esaki, Y. Katsube, K. Iwamura and T. Kodama, "A Study on Connection Admission Control for an ATM Network," Technical Report, R & D Center, Toshiba Corp., 1989.

[ELL89] A. E. Eckberg, Jr., D. T. Luan and D. M. Lucantoni, "Meeting the Challenge: Congestion and Flow Control Strategies for Broadband Information Transport," Proc. IEEE GLOBECOM '89, pp.49.3.1-49.3.5.

[FRAN87] V. Frantzen, "Trends in the Development of Public Telecommunication Networks," Computer Networks and ISDN Syst., Vol.14, pp.339-358, 1987.

[GAC86] P. Gonet, P. Adams and J. Coudreuse, "Asynchronous Time-Division Switching: The Way to Flexible Broadband Communication Networks," IEEE Int. Zurich Seminar on Digital Communications, March 1986.

[GHAN89] M. Ghanbari, "Two-Layer Coding of Video Signals for VBR Networks," IEEE J. Select. Areas Commun., Vol.7, No.5, pp.771-781., June 1989.

[GL89] A. Gersht and K. J. Lee, "A Congestion Control Framework for ATM Networks," Proc. IEEE INFOCOM '89, pp.701-710.

[GOLD77] H. Goldberg, "Analysis of the Earliest Due Date Scheduling Rule in Queueing Systems," Mathematics of Operations Research, Vol.2, No.2, pp.145-154, May 1977.

[GOOD80] D. J. Goodman, "Embedded DPCM for variable bit rate transmission," IEEE Trans. Commun., Vol.COM-28, pp.1040-1046, July 1980.

[GRF89] G. Gallassi, G. Rigolio and L. Fratta, "ATM: Bandwidth Assignment and Bandwidth Enforcement Policies," Proc. IEEE GLOBECOM '89, pp.49.6.1-49.6.6.

[HA87] J. Y. Hui and E. Arthurs, "A Broadband Packet Switch for Integrated Transport," IEEE J. Select. Areas Commun., SAC-5, No.8, pp.1264-1273, October 1987.

[HAND89] R. Handel, "Evolution of ISDN Towards Broadband ISDN," IEEE Network, pp.7-13, January 1989.

[HIRA89] A. Hiramatsu, "ATM Communications Network Control by Neural Network," Intl. Joint Conf. on Neural Networks '89, Wash. DC, June 1989.

[HL86] H. Heffes and D. Lucantoni,"A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance," IEEE J. Select. Areas Commun., Vol.SAC-4, No.6, pp.856-868, September 1986.

[HM89] A. Hac and H. B. Mutlu, "Synchronous Optical Network and Broadband ISDN Protocols," Computer, Vol.22, No.11, pp.26-34, November 1989.

[HOBE83] W. L. Hoberecht, "A Layered Network Protocol for Packet Voice and Data Integration," IEEE J. Select. Areas Commun., Vol.SAC-1, No.6, December 1983.

[HOLT89] J. M. Holtzman, "Coping with Broadband Traffic Uncertainties: Statistical Uncertainty, Fuzziness, Neural Networks," presented at IEEE Workshop on Computer Commun., Data Point, CA, October 1989.

[HP91] N. C. Hutchinson and L. L. Peterson, "The x-Kernel: An Architecture for Implementing Network Protocols," IEEE Transactions on Software Engineering, January 1991.

[HUNT83] J. J. Hunter, *Mathematical Techniques of Applied Probability*, Vol.II, Academic Press, New York, pp.193, 1983.

[HW89] M. Hirano and N. Watanabe, "Characteristics of a Cell Multiplexer for Bursty ATM Traffic," Proc. IEEE ICC '89, pp.13.2.1-13.2.5.

[IAU87] H. Ichikawa, M. Aoki and T. Uchiyama, "High-Speed Packet Switching Systems for Multimedia Communications," IEEE J. Select. Areas in Commun., Vol.SAC-5, No.8, pp.1336-1345, October 1987.

[IDE88] I. Ide, "Superposition of Interrupted Poisson Processes and Its Application to Packetized Voice Multiplexers," Proc. 12th Intl. Teletraffic Congress, Torino, Italy, 1988.

[IP80] M. Irland and G. Pujolle, "Comparison of Two Packet Retransmission Techniques," IEEE Trans. Inform. Theory, Vol.IT-26, pp.92-97, 1980.

[JACK60] J. R. Jackson, "Some Problems in Queueing with Dynamic Priorities," Naval Res. Log. Quart., Vol.7, No.3, 1960.

[JACK61] J. R. Jackson, "Queues with Dynamic Priority Discipline," Management Science, Vo.8, No.1, 1961.

[JACK62] J. R. Jackson, "Waiting Time Distributions for Queues with Dynamic Priorities," Naval Res. Log. Quart., Vo.9. No.1, 1962.

[JACK63] J. R. Jackson, "Jobshop-like Queueing Systems," Management Science 10, 1963.

[JC81] N. S. Jayant and S. W. Christensen, "Effects of packet losses in waveform codded speech and improvements due to an odd-even sample-interpolation procedure," IEEE Trans. Commun., Vol. COM-29, pp.101-109, February 1981.

[JMDS90] S. B. Jacobsen, K. Moth, L. Dittmann and K. Sallberg, "Load Control in ATM Networks," submitted to ISS '90.

[KM84] J. Kulzer and W. Montgomery, "Statistical Switching Architecture for Future Services," Proc. ISS '84, pp.43A.1.1-43A.1.6.

[KMHY89] F. Kishino, K. Manabe, Y. Hayashi and H. Yasuda, "Variable Bit-Rate Coding of Video Signals for ATM Networks," IEEE J. Select. Areas Commun., Vol.7, No.5, pp.801-806, June 1989.

[KRON90] H. Kroner, "Comparative Performance Study of Space Priority Mechanisms for ATM Networks," Proc. IEEE INFOCOM '90, pp.1136-1143.

[KS89] T. Kamitake and T. Suda, "Evaluation of an Admission Control Scheme for an ATM Network Considering Fluctuations in Cell Loss Rate," Proc. IEEE GLOBECOM '89, pp.49.4.1-49.4.7.

[KSB88] V. R. Karanam, K. Sriram and D. O. Bowker, "Performance Evaluation of Variable-Bit-rate Voice in Packet-Switched Networks," AT&T Tech. J., pp.57-69, September/October, 1988.

[KUHL83] D. Kuhl, "Error Recovery Protocols: Link-by-Link versus Edge-to-Edge," Proc. IEEE INFOCOM '83, pp.319-324.

[LAM76] S. S. Lam, "Store and Forward Buffer Requirements in a Packet Switching Network," IEEE Trans. Commun., Vol. COM-24, pp.394-403, April 1976.

[LI89a] S. Q. Li, "Overload Control in a Finite Message Storage Buffer," IEEE Trans. on Commun., Vol.37, No.12, pp.1330-1338, December 1989.

[LI89b] S-Q. Li, "Study of Information Loss in Packet Voice Systems," IEEE Trans. Commun., Vol.37, No.11, pp.1192-1202, November 1989.

[LK88] Y. Lim and J. Kobza, "Analysis of a Delay-Dependent Priority Discipline in a Multi-Class Traffic Packet Switching Node," Proc. IEEE INFOCOM '88, pp.9A.4.1-9A.4.1.10.

[LL89] R-J. Li and E. S. Lee, "Analysis of a Fuzzy Queue," Computers Math & Applic., Vol.17, No.7, pp.1143-1147, 1989.

[LM89] K. Q. Liao and L. G. Mason, "A Discrete-Time Single Server Queue With a Two-Level Modulated Input and Its Applications," Proc. IEEE GLOBECOM '89, pp.26.1.1-26.1.6.

[LP90] D. M. Lucantoni and S. P. Parekh, "Selective Cell Discard Mechanisms for a B-ISDN Congestion Control Architecture," Proc. 7th Intl. Teletraffic Congress Seminar, October 1990.

[LS78] R. S. Liptser and A. N. Shiryaev, *Statistics of Random Processes*, Vols.I,II, Springer-Verlag, New York, 1978.

[MASKR88] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson and J. D. Robbins, "Performance Models of Statistical Multiplexing in Packet Video Communications," IEEE Trans. Commun., Vol.36, No.7, pp.834-844, July 1988.

[MEIE89] K. S. Meier-Hellstern, "The Analysis of a Queue Arising in Overflow Models," IEEE Trans. Commun., Vol.37, No.4, pp.367-372, April 1989.

[MINZ87] S. E. Minzer, "Broadband User-Network Interfaces to ISDN," Proc. IEEE ICC '87, pp.11.2.1-11.2.6.

[MINZ89] S. E. Minzer, "Broadband ISDN and Asynchronous Transfer Mode (ATM)," IEEE Communications Magazine, Vol.27, No.9, pp.17-24, September 1989.

[MOSM89] M. Murata, Y. Oie, T. Suda and H. Miyahara, "Analysis of a Discrete-Time Single-Server Queue with Bursty Inputs for Traffic Control in ATM Networks," Proc. IEEE GLOBECOM '89, pp.49.5.1-49.5.7.

[MSZ86] R. W. Muise, T. J. Schoenfeld and G. H. Zimmerman, "Experiments with Wideband Packet Technology," Proc. 1986 Intl. Zurich Seminar Digital Commun., paper D4, March 1986.

[MT87] M. Murata and H. Takagi, "Two-Layer Modeling for Local Area Networks," Proc. IEEE INFOCOM '87, pp.132-140.

[MULL90] S. J. Mullender, *Distributed Systems*, ACM Press, 1990.

[MW89] K. S. Meier-Hellstern and P. E. Wirth, "Packet Switch Provisioning Accounting for uncertainty in the Forecasting of Customer Characteristics," Proc. 7th Intl. Teletraffic Congress Specialists Seminar, Adelaide, Australia, September 1989.

[NEUT89] M. F. Neuts, *Structured Stochastic Matrices of M/G/1 Type And Their Applications*, Marcel Dekker, Inc., New York, 1989.

[NFO89] M. Nomura, T. Fujii and N. Ohta, "Basic Characteristics of Variable Rate Video Coding in ATM Environment," IEEE J. Select. Areas Commun., Vol.7, No.5, pp.752-760, June 1989.

[NKN89] K. Nakamaki, M. Kawakatsu and A. Notoya, "Traffic Control for ATM Networks," Proc. IEEE ICC '89, pp.22.5.1-22.5.5.

[NKT91] R. Nagarajan, J. F. Kurose and D. Towsley, "Approximation Techniques for Computing Packet Loss in Finite-Buffered Voice Multiplexers," IEEE J. Select. Areas Commun., Vol. 9, No. 3, pp. 368-377, 1991.

[NL90] A. A. Nilsson and F-Y. Lai, "Performance Evaluation of Error Recovery Schemes in High Speed Networks," Proc. IEEE ICC '90, pp.315.2.1-315.2.5.

[OST88] S. Ohta, K. Sato, and I. Tokizawa, "A Dynamically Controllable ATM Transport Network Based on the Virtual Path Concept," Proc. IEEE GLOBECOM '88, pp.39.2.1-39.2.5.

[PAPO84] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, New York: McGraw Hill, 1984.

[PDF89] D. W. Petr, L. A. DaSilva, Jr., and V. S. Frost, "Priority discarding of Speech in Integrated Packet Networks," IEEE J. Select. Areas Commun., Vol.7, No.5, pp.644-659, June 1989.

[PF91] D. W. Petr and V. S. Frost, "Nested Threshold Cell Discarding for ATM Overload Control: Optimization Under Cell Loss Constraints," Proc. IEEE INFOCOM'91, pp.12A.4.1-12A.4.10.

[PRIN89] G. L. Pringle, "SONET: problem or opportunity?," Telephony Magazine, pp.61-65, August 1989.

[PRYC91] M. D. Prycker, *Asynchronous Transfer Mode: Solution for Broadband ISDN*, Ellis Horwood, 1991.

[PS75] M. C. Pennotti and M. Schwartz, "Congestion Control in Store and Forward Tandem Links," IEEE Trans. Commun., Vol.COM-23, December 1975.

[PTW88] S. Panwar, D. Towsley and J. Wolf, "Optimal Scheduling Policies for a Class of Queues with Customer Deadlines to the Beginning of Service," J. ACM, Vol.35, No.4, pp.832-844, October 1988.

[RIDE88] M. J. Rider, "Protocols for ATM Access Networks," Proc. IEEE GLOBECOM '88, pp.4.4.1-4.4.6.

[RS] W. A. Rosenkrantz and R. Simha, "Some Theorems on Conditional Pasta: A Stochastic Integral Approach," Operations Research Letters, to appear.

[SC81] C. H. Sauer and K. M. Chandy, "Computer Systems Performance Modeling," Prentice-hall, 1981.

[SL76] P. J. Schweitzer and S. S. Lam, "Buffer Overflow in a Store-and-Forward Network Node," IBM J. Res. Develop., pp.542-550, November 1976.

[SL88] K. Sriram and D. M. Lucantoni, "Traffic Smoothing Effects of Bit Dropping In a Packet Voice Multiplexer," Proc. IEEE INFOCOM '88, pp.8A.1.1-8A.1.12.

[SLCG89] M. Sidi, W. Z. Liu, I. Cidon and I. Gopal, "Congestion Control Through Input Rate Regulation," Proc. IEEE GLOBECOM '89, pp.49.2.1-49.2.5.

[SMRA89] P. Sen, B. Maglaris, N. E. Rikli and D. Anastassiou, "Models for Packet Switching of Variable-Bit-Rate Video Sources," IEEE J. Select. Areas Commun., Vol.7, No.5, pp.865-869, June 1989.

[SW86] A. Sriam and W. Whitt, "Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data," IEEE Journal on Selected Areas in Communications, Vol.SAC-4, No.6, pp.833-846, September 1986.

[TST89] K. Toyoshima, M. Sasagawa and I. Tokizawa, "Flexible Surveillance Capabilities for ATM-based Transmission Systems," Proc. IEEE ICC '89, pp.22.3.1-22.3.6.

[TURN86] J. S. Turner, "New Directions in Communications (or Which way to the Information Age?)," IEEE Commun. Magazine, Vol.25, No.10, pp.8-15, October 1986.

[VICK90] R. Vickers, Broadband ISDN Tutorial #3, IEEE INFOCOM'90.

[VPV88] W. Verbiest, L. Pinnoo and B. Voeten, "The Impact of the ATM Concept on Video Coding," IEEE J. Select. Areas Commun., Vol.6, No.9, pp.1623-1632, December 1988.

[VV88] J. P. Vorstermans and A. P. De Vleeschouwer, "Layered ATM Systems and Architectural Concepts for Subscribers' Premises Networks," IEEE J. Select. Areas Commun., Vol.6, No.9, pp.1545-1555, December 1988.

[WHIT83] W. Whitt, "The Queueing Network Analyzer," Bell Syst. Tech. J., Part 1, Vol.62, No.9, pp.2779-2815, 1983.

[WK90] G. M. Woodruff and R. Kositpaiboon, "Multimedia Traffic Management Principles for Guaranteed ATM Network Performance," IEEE J. Select. Areas Commun., Vol.8, No.3, pp.437-446, April 1990.

[WONG78] J. W. Wong, "Queueing Network Modeling of Computer Communication Networks," ACM Computing Surveys, Vol.10, No.3, September 1978.

[WRR88] G. M. Woodruff, R. G. Rogers, and P. S. Richards, "A Congestion Control Framework for High-Speed Integrated Packetized Transport," Proc. IEEE GLOBECOM '88, pp.7.1.1-7.1.5.

[YLS87] N. Yin, S-Q. Li and T. E. Stern, "Congestion Control for Packet Voice by Selective Packet Discarding," Proc. IEEE GLOBECOM '87, pp.45.3.1-45.3.4.

[YLS88] N. Yin, S-Q. Li and T. E. Stern, "Data Performance in an Integrated Packet Voice/Data System Using Voice Congestion Control," Proc. IEEE GLOBECOM '88, pp.16.4.1-16.4.5.

[YMKM89] H. Yamada, K. Miyake, F. Kishino and K. Manabe, "Modeling of Arrival Process of Packetized Video and Related Statistical Multiplexer Performance," Proc. IECEJ National Conference, 1989.

[YS91] H. Yamada and S. Sumita, "A Traffic Measurement Method and its Application for Cell Loss Probability Estimation in ATM Networks," IEEE J. Select. Areas Commun., Vol.SAC-9, No.3, pp.315-324, April 1991.

[YSL87] N. Yin, T. E. Stern and S-Q. Li, "Performance Analysis of a Priority-Oriented Packet Voice System," Proc. IEEE INFOCOM '87, pp.856-863.

[YYOK89] Y. Yasuda, H. Yasuda, N. Ohta and F. Kishino, "Packet Video Transmission Through ATM Networks," Proc. IEEE GLOBECOM '89, pp.25.1.1-25.1.5.

Appendix A

Discussions on the Analytic Model in Chapter 3

In our analytic model used in Chapter 3, we introduced two major assumptions: the Poisson assumption (i.e., the generation of new packets follows a Poisson process) and the exponential protocol processing time assumption (i.e., the protocol-processing times at each layer follow an exponential distribution.) Justification is now given for these assumptions.

Poisson Assumption

In the analytical model, we assumed that the generation of new packets follows a Poisson process: the new packet arrivals at PT-layer from the upper layer follow a Poisson process. This is because we primarily focused on data applications and investigated the effects of error recovery schemes. A Poisson process is widely accepted as an arrival process suitable to represent data traffic. It is generally agreed that the real time traffic, such as voice and video with layered coding, may not require strict error control. Such traffic is robust to the packet loss, and in addition, retransmissions due to error control may prevent the real time delivery. Error recovery schemes such as those investigated in Chapter 3 are useful mainly for data applications. Therefore, we have focused primarily on data applications and modeled a packet generation process as a Poisson process.

In addition to high-speed networking, we are also interested in frame relaying, a technique for data packet transmission used in N-ISDN. CCITT recommends frame relaying to support data packet transmissions over B/D channels of N-ISDN [PRYC91]. Frame relaying is based on the X.25 packet switching, but has less functionality than X.25. In the frame relaying network, retransmission of user data frames, as needed for error correction, are only performed end-to-end, and thus, a higher throughput can be achieved. A Poisson process describes the data traffic on N-ISDN, and our analysis is directly applicable to the frame relaying.

Although assuming a Poisson process for new packet arrivals is suitable for our purpose (as explained above) and is consistent with the assumption made in the past work, [1] it is somewhat restrictive to represent multimedia traffic. Some research show that multimedia traffic in high-speed networks may not be accurately modeled by a Poisson process, nor by any renewal process [YS91, HL86, SW86]. Accurate modeling of multimedia traffic is an emerging and important research area, and assuming a non-renewal input process makes the analysis of the network-of-queues a challenge. The next step of the research presented in Chapter 3 is to incorporate multimedia traffic into the analytical model.

[1] The Poisson assumption has been commonly made in the past work on the investigation of error control schemes. In fact, in all previous studies [BKTV88, BS90a, NL90] that investigate error control schemes in the setting of a high-speed network, a Poisson process is used to represent new packet arrivals. Also, new packet arrivals have been modeled as a Poisson process in past research (e.g., [LAM76, IP80, BRAD88, SL76]), which investigate error control schemes in data applications.

Exponential Protocol-Processing Time

In the analysis, it is assumed that the protocol-processing time at each layer follows an exponential distribution. The following section gives justification for assuming the exponential protocol-processing time at each layer.

It is widely recognized that the protocol-processing performance bottleneck lies in both the protocol itself and the operating system that supports the protocol's execution (e.g., process scheduling, data copying, buffer management, and timer management) [HP91, MULL90, CJRS89]. In our work, we focus on the combined overhead of both the protocol and the operating system. This combined overhead is referred to as the protocol-processing overhead in our work, and the time that packets spend due to this overhead is called the protocol-processing time. In our analytical model, service at each protocol layer represents this protocol/OS combined overhead at that layer.

Various factors affect the protocol/OS combined overhead. To begin with, there are many possible paths that packets take through the code implementing a protocol [CJRS89]. For instance, connection management, error detection, and error recovery results in executing special code. However, most of this code is not required for the normal data transfer case. This results in random processing times for different packets.

When sending/receiving packets, it is often required to find the local state information [COME91]. The time required to search and retrieve the state information from a state table depends on, for instance, whether the segment that contains the requested information currently resides in a cache or in a main memory. If it resides in a cache, it takes less time to retrieve such information [CJRS89, ABBGRTY86]. This memory read time is random. Other factors that affect the read time include data structures and algorithms for search. For example, storing connection information in a linked list, e.g., certain BSD UNIX TCP/IP implementations, results in worse demultiplexing performance as the number of open connections increases [HP91].

If the communication protocol is implemented on a uni-processor machine, a communication process may be suspended and swapped out due to an interrupt from another process. (The same situation can happen on a multi-processor machine running more logical processes compared with the pool of available physical processors.) In addition, asynchronous events within a single process can also increase overhead, e.g., handling timer expiration when packets time out. This overhead time is random.

As illustrated above, the protocol-processing time at each layer depends on a variety of factors and is random. It is also reasonable to assume that the processing time of one packet is drawn independently of the processing time of the other packets. From these observations, the we conclude that assuming exponential protocol-processing times is acceptable.

One consideration is the protocol-processing time at a sender L-layer queue. In our queueing model, this includes, among others, the packet transmission time on a physical channel (the packet length divided by the channel speed). The packet length distribution may seem to affect the distribution of the processing time at a sender L-layer. However, the major components comprising the service at a L-layer queue are the time to perform data link protocol functions (such as framing, error checking, and addressing) and the associated operating system overhead (to perform functions such as timer management and buffer management). Time required to perform the above functions is an order of magnitude larger than the packet transmission time in a typical high-speed networking environment [CJRS89]. Therefore, the assumption of the exponential protocol-processing time is reasonable.

Appendix B

Individual Loss Probabilities for Multiple Stream Inputs (Continuous-Time Case)

In this appendix, we extend the continuous-time analysis presented in Section 4.1 to the case in which more than two heterogeneous input streams are multiplexed. We obtain the packet loss probability for each input stream. The same analytical techniques used in Section 4.1 can be applied for a multiple stream input case. In the following, we denote the number of input streams as N, and the maximum buffer size as $K-1$ (i.e., the maximum system size of K).

The arrival process of each input stream is assumed to be a 2-state MMPP. The "driving" states of each MMPP are labeled 1 and 2. For the stream i, the transition rate from state 1 to state 2 is denoted by α_i, and the transition rate from state 2 to state 1 is denoted by β_i. The packet arrivals when the stream i is in state 1 follow a Poisson process with rate $\lambda_{1,i}$, whereas the packet arrival rate in state 2 is $\lambda_{2,i}$. The service times of packets are exponentially distributed with rate μ.

Let $N_i(t)$ denote the cumulative number of arrivals from the i^{th} stream in the time interval $[0,t]$, and $U_K(t)$ denote the indicator function for the system state K at time $t-$ (i.e., $U_K(t) = 1$, iff $Z(t-) = K$). Let $P_{loss}(i)$ denote the long-term loss probability for the i^{th} stream. Then, from Eq.(4.6), we have

$$P_{loss}(i) = \lim_{t\to\infty} \frac{t}{N_i(t)} \lim_{t\to\infty} \frac{1}{t} \int_0^t U_K(s)dN_i(s). \tag{B.1}$$

Also, from Eq.(4.7), we have

$$\lim_{t\to\infty} \frac{N_i(t)}{t} = \frac{\lambda_{1,i}\beta_i + \lambda_{2,i}\alpha_i}{\alpha_i + \beta_i}, \tag{B.2}$$

and from Eq.(4.9), we obtain

$$\lim_{t\to\infty} \frac{1}{t} \int_0^t U_K(s)dN_i(s) = \lambda_{1,i}\pi(1_i, K) + \lambda_{2,i}\pi(2_i, K) \tag{B.3}$$

Here, $\pi(1_i, K)$ and $\pi(2_i, K)$ are given by the following:

$$\pi(1_i, K) = \sum_{j_1=1}^{2} \sum_{j_2=1}^{2} \cdots \sum_{j_N=1}^{2} \pi(j_1, j_2, \cdots, j_{i-1}, 1, j_{i+1}, \cdots, j_N, K)$$

$$\pi(2_i, K) = \sum_{j_1=1}^{2} \sum_{j_2=1}^{2} \cdots \sum_{j_N=1}^{2} \pi(j_1, j_2, \cdots, j_{i-1}, 2, j_{i+1}, \cdots, j_N, K)$$

By substituting Eqs.(B.2) and (B.3) into Eq.(B.1), we can easily obtain the packet loss probability for stream i, $P_{loss}(i)$. For the overall packet loss probability, $P_{loss}(O)$, from Eq.(4.12), we have

$$P_{loss}(O) = \lim_{t\to\infty} \frac{t}{N(t)} \lim_{t\to\infty} \frac{1}{t} \int_0^t U_K(s)dN(s) \tag{B.4}$$

Since $N(t) = \sum_{i=1}^{N} N_i(t)$, we have

$$\begin{aligned} \lim_{t\to\infty} \frac{N(t)}{t} &= \sum_{i=1}^{N} \lim_{t\to\infty} \frac{N_i(t)}{t} \\ &= \sum_{i=1}^{N} \frac{\lambda_{1,i}\beta_i + \lambda_{2,i}\alpha_i}{\alpha_i + \beta_i}, \end{aligned} \tag{B.5}$$

and

$$\begin{aligned} \lim_{t\to\infty} \frac{1}{t} \int_0^t U_K(s)dN(s) &= \sum_{i=1}^{N} \lim_{t\to\infty} \frac{1}{t} \int_0^t U_K(s)dN_i(s) \\ &= \sum_{i=1}^{N} \{\lambda_{1,i}\pi(1_i, K) + \lambda_{2,i}\pi(2_i, K)\}. \end{aligned} \tag{B.6}$$

By substituting Eqs.(B.5) and (B.6) into Eq.(B.4), we can easily obtain the overall loss probability.

To solve for the limiting probability $\pi(j_1, j_2, \cdots, j_N, q)$, we again use a direct numerical solution. We represent the system state by $(j_1, j_2, \cdots, j_N, q)$, where j_i $(j_i = 1, 2)$ is the state of the i^{th} stream $(i = 1, 2, \cdots, N)$, and q is the number of packets in the system. Then, for $i, k \in \{1, 2, \cdots, N\}, j_i, j'_i \in \{1, 2\}$, $q, q' \in \{0, 1, 2, ..., K\}$, the infinitesimal generator $\mathbf{Q}$, is given by

$$\mathbf{Q}_{(j_1,j_2,\cdots,j_N,q)\rightarrow(j'_1,j'_2,\cdots,j'_N,q')} =$$
$$\begin{cases} r_{j_i j'_i}, & \text{if } j'_i \neq j_i, j'_k = j_k \text{ for all } k \neq i, q' = q \\ \sum_{i=1}^{N} \lambda_{j_i,i}, & \text{if } j'_i = j_i \text{ for all } i, q' = q+1 \\ \mu, & \text{if } j'_i = j_i \text{ for all } i, q' = q-1 \\ -\sum_{i=1}^{N} r_{j_i j_{\bar{i}}} - \sum_{i=1}^{N} \lambda_{j_i,i}, & \text{if } j'_i = j_i \text{ for all } i, q' = q = 0 \\ -\mu - \sum_{i=1}^{N} r_{j_i j_{\bar{i}}}, & \text{if } j'_i = j_i \text{ for all } i, q' = q = K \\ -\mu - \sum_{i=1}^{N} r_{j_i j_{\bar{i}}} - \sum_{i=1}^{N} \lambda_{j_i,i}, & \text{if } j'_i = j_i \text{ for all } i, q' = q, 0 < q < K \\ 0, & \text{otherwise} \end{cases} \quad \text{(B.7)}$$

where $r_{j_i\bar{j}'_i}$ is the transition rate of an arrival process from state j_i to state j'_i, and $\bar{i}$ represents the complementary state of i (i.e., $\bar{1} = 2, \bar{2} = 1$). $r_{j_i j'_i}$ is given by $r_{j_i j'_i} = \begin{cases} \alpha_i, & \text{if } j_i = 1, j'_i = 2 \\ \beta_i, & \text{if } j_i = 2, j'_i = 1 \end{cases}$.

Appendix C

Individual Loss Probabilities for Multiple Stream Inputs (Discrete-Time Case)

In this appendix, we extend the discrete-time analysis presented in Section 4.2 to the case in which more than two heterogeneous input streams are multiplexed. We obtain the packet loss probability for each input stream. In the following, we denote the number of input streams as N, and the maximum buffer size as $K-1$ (i.e., the maximum system size of K).

The arrival process of each input stream is assumed to be a 2-state MMBP. The "driving" states of each MMBP are labeled 1 and 2. For the stream i, transition rates between states are α_i (from state 1 to state 2) and β_i (from state 2 to state 1). Packets arrive according to a Bernoulli process, and for the stream i, the probability of having an arrival in a slot is $p_{1,i}$ when it is in state 1 and $p_{2,i}$ when it is in state 2. Service times are geometrically distributed, and the probability of service completion in a slot, provided the server is busy, is s.

We focus on the i^{th} stream, and we define the following indicator functions for each slot n to obtain the packet loss probability for the i^{th} stream:

- $J_l(n)$: $J_l(n) = 1$ *iff* l number of arrivals have taken place from the remaining $N-1$ streams in the n^{th} slot. Note that, when $J_l(n) = 1$, a total of $l+1$ arrivals (including the arrival from the i^{th} stream) have taken place in the n^{th} slot.
- $V_{l,q}(n)$: $V_{l,q}(n) = 1$ *iff* in the n^{th}slot, a packet from the i^{th} stream is discarded when a total of l packets arrive at the system state q.

Let $U(n)$ denote the indicator function for the state in which a packet from the i^{th} stream is discarded in the n^{th} slot. Then, we have

$$\begin{aligned} U(n) &= U_K(n) \\ &+U_{K-1}(n)\{J_1(n)V_{2,K-1}(n) + J_2(n)V_{3,K-1}(n) + \cdots + J_{N-1}(n)V_{N,K-1}(n)\} \\ &+U_{K-2}(n)\{J_2(n)V_{3,K-2}(n) + J_3(n)V_{4,K-2}(n) + \cdots + J_{N-1}(n)V_{N,K-2}(n)\} \\ &\vdots \\ &+U_{K-N+1}(n)J_{N-1}(n)V_{N,K-N+1}(n). \end{aligned} \tag{C.1}$$

For the random packet discarding scheme, we have

$$P[V_{l,q}(n) = 1] = \lim_{m\to\infty} \frac{1}{m} \sum_{n=0}^{m} V_{l,q}(n) = \frac{l-(K-q)}{l} \tag{C.2}$$

This is because, out of l packets, only $K-q$ number of packets are accepted and the remaining $l-(K-q)$ number of packets are lost. For the remaining derivation, the same analytical technique used in section 4.2.2 can apply.

As in section 4.2.2, a direct numerical method can be used to solve for the limiting probabilities. The transition probability matrix **P** is given in Figure C.1. In Figure C.1, the matrix **B** is given by the Kronecker product

$$\mathbf{B} = \mathbf{B_1} \otimes \mathbf{B_2} \otimes \cdots \otimes \mathbf{B_N} \tag{C.3}$$

where

$$\mathbf{B_i} = \begin{pmatrix} 1-\alpha_i & \alpha_i \\ \beta_i & 1-\beta_i \end{pmatrix}. \tag{C.4}$$

$$
\begin{array}{c|ccccccccccc}
 & 0 & 1 & 2 & \cdots & N & N+1 & \cdots & K-2 & K-1 & K \\
\hline
0 & (P_{1a}P_{1d}+P_{0a})B & (P_{1a}P_{0d}+P_{2a}P_{1d})B & (P_{2a}P_{0d}+P_{3a}P_{1d})B & \cdots & P_{Na}P_{0d}B & & & & & \\
1 & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{1a}P_{1d})B & (P_{1a}P_{0d}+P_{2a}P_{1d})B & \searrow & \cdots & P_{Na}P_{0d}B & & & & \\
2 & \searrow & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{1a}P_{1d})B & & & \searrow & & & & \\
\cdot & & \searrow & & \searrow & & \cdots & & & & \\
\cdot & & & & & \vdots & & & & & \\
\cdot & & & & & & & & & & \\
K-N+1 & & & & & \cdots & & & (P_{(N-3)a}P_{0d}+P_{(N-2)a}P_{1d})B & (P_{(N-2)a}P_{0d}+P_{[\geq(N-1)a]}P_{1d})B & P_{[\geq(N-1)a]}P_{0d}B \\
K-N+2 & & & & & & \cdots & & (P_{(N-4)a}P_{0d}+P_{(N-3)a}P_{1d})B & (P_{(N-3)a}P_{0d}+P_{[\geq(N-2)a]}P_{1d})B & P_{[\geq(N-2)a]}P_{0d}B \\
\cdot & & & & & & & & & & \\
\cdot & & & & & \vdots & & & & & \\
\cdot & & & & & & & & & & \\
K-1 & & & & & & & & P_{0a}P_{1d}B & (P_{0a}P_{0d}+P_{0a}P_{1d})B & P_{0a}P_{0d}B \\
K & & & & & & & & & P_{1d}B & P_{0d}B
\end{array}
$$

Figure C.1: Transition Probability Matrix for Multiple Stream Case (Discrete Time)

$P_{\{\geq ia\}}$ in Figure C.1 represents the probability that at least i number of packets arrive. For the rest of the notation in Figure C.1, same definition used in section 4.2.2 is used.

Appendix D

Individual Loss Probabilities for Multiple Stream Inputs (Priority Packet Discarding Scheme)

In this appendix, we extend the analysis of a priority packet discarding scheme presented in Section 4.3 to the case in which more than two heterogeneous input streams are multiplexed. N heterogeneous sources are multiplexed into a single first-come-first-served queue whose maximum capacity is $K-1$ (i.e., the maximum system size is K). Since the analysis for the continuous-time case is a straightforward extension of the analysis presented in the section 4.3.1, only the discrete-time case is considered here.

Again, a threshold-based priority discarding scheme is considered. It is assumed that if the current system occupancy is less than the threshold, no low priority packets are discarded even if accepting all the low priority packets exceeds the threshold, as opposed to only accepting low priority packets up to the threshold.

Let q denote the system occupancy and θ denote the threshold. For the arrival process of each stream, refer to Appendix C. Assume that among N streams, H number of streams are high priority streams, and the rest of the streams are low priority streams. For the i^{th} stream, again, let $U(n)$ denote the indicator function for the state in which the stream i packet is discarded in the n^{th} slot, and define the following indicator functions for each slot n:

- $J_l(n)$: $J_l(n) = 1$ iff l number of arrivals have taken place from the remaining $N-1$ streams in the n^{th} slot. Note that, when $J_l(n) = 1$, a total of $l+1$ (including the arrival from the i^{th} stream) number of arrivals have taken place in the n^{th} slot.
- $J_l^h(n)$: $J_l^h(n) = 1$ iff l number of arrivals have taken place from the remaining $H-1$ high priority streams. Note that, when $J_l(n) = 1$, a total of $l+1$ (including the arrival from the i^{th} stream) number of arrivals have taken place from the high priority streams in the n^{th} slot.
- $V_{l,q}(n)$: $V_{l,q}(n) = 1$ iff in the n^{th} slot, the i^{th} stream packet is discarded when a total of l number of arrivals occur at the system state q.

First, let us consider the case when $\theta \leq K - N + 1$. Here, no packet will be discarded when $q < \theta$. When the stream i is a low priority stream,

$$U(n) = \sum_{q=\theta}^{K} U_q(n) \tag{D.1}$$

since arriving low priority packets are always discarded when the queue occupancy is greater than or equal to θ. When the stream i is a high priority stream,

$$\begin{aligned} U(n) &= U_K(n) \\ &\quad + U_{K-1}(n)\{J_1^h(n)V_{2,K-1}(n) + J_2^h(n)V_{3,K-1}(n) + \cdots + J_{H-1}^h(n)V_{H,K-1}(n)\} \\ &\quad + U_{K-2}(n)\{J_2^h(n)V_{3,K-2}(n) + J_3^h(n)V_{4,K-2}(n) + \cdots + J_{H-1}^h(n)V_{H,K-2}(n)\} \\ &\quad \vdots \\ &\quad + U_{K-H+1}(n)J_{H-1}^h(n)V_{H,K-H+1}(n). \end{aligned} \tag{D.2}$$

When $q = K$, arriving high priority packets are always discarded, and when $K - H + 1 \leq q \leq K - 1$, arriving high priority packets are discarded only if the number of arrivals from high priority streams exceeds the available buffer space. We assume that when the number of arrivals from high priority streams exceeds the available buffer space, the packets to be discarded are randomly picked. Therefore,

$$P[V_{l,q}(n) = 1] = \lim_{m \to \infty} \frac{1}{m} \sum_{n=0}^{m} V_{l,q}(n) = \frac{l - (K - q)}{l} \tag{D.3}$$

since out of l packets, only $K - q$ number of packets will be accepted and the remaining $l - (K - q)$ number of packets will be discarded.

Now, let us consider the case when $\theta > K - N + 1$. In this case, when $K - N + 1 \leq q < \theta$, if the number of arrivals exceeds the available buffer space, packets will be discarded randomly. Therefore, when the stream i is a low priority stream,

$$\begin{aligned} U(n) &= \sum_{q=\theta}^{K} U_q(n) \\ &\quad + U_{\theta-1}(n)\{J_{K-\theta+1}(n)V_{K-\theta+2,\theta-1}(n) + J_{K-\theta+2}(n)V_{K-\theta+3,\theta-1}(n) \\ &\quad + \cdots + J_{N-1}(n)V_{N,\theta-1}(n)\} \\ &\quad + U_{\theta-2}(n)\{J_{K-\theta+2}(n)V_{K-\theta+3,\theta-2}(n) + J_{K-\theta+3}(n)V_{K-\theta+4,\theta-2}(n) \\ &\quad + \cdots + J_{N-1}(n)V_{N,\theta-2}(n)\} \\ &\quad \vdots \\ &\quad + U_{K-N+1}(n)J_{N-1}(n)V_{N,K-N+1}(n). \end{aligned} \tag{D.4}$$

When $q \geq \theta$, arriving low priority packets are always discarded (the first line on the right hand side), and when $K - N + 1 \leq q < \theta$, arriving packets are discarded randomly when the number of arrivals exceeds the available buffer space (the rest of the terms on the right hand side).

When the stream i is a high priority stream, we need to consider the following two cases: (1) $\theta \leq K-H+1$ and (2) $\theta > K-H+1$. For the case when $\theta \leq K-H+1$,

$$
\begin{aligned}
U(n) \;=\; & U_K(n) \\
& +U_{K-1}(n)\{J_1^h(n)V_{2,K-1}(n) + J_2^h(n)V_{3,K-1}(n) + \cdots + J_{H-1}^h(n)V_{H,K-1}(n)\} \\
& +U_{K-2}(n)\{J_2^h(n)V_{3,K-2}(n) + J_3^h(n)V_{4,K-2}(n) + \cdots + J_{H-1}^h(n)V_{H,K-2}(n)\} \\
& \vdots \\
& +U_{K-H+1}(n)J_{H-1}^h(n)V_{H,K-H+1}(n) \\
& +U_{\theta-1}(n)\{J_{K-\theta+1}(n)V_{K-\theta+2,\theta-1}(n) + J_{K-\theta+2}(n)V_{K-\theta+3,\theta-1}(n) \\
& + \cdots + J_{N-1}(n)V_{N,\theta-1}(n)\} \\
& +U_{\theta-2}(n)\{J_{K-\theta+2}(n)V_{K-\theta+3,\theta-2}(n) + J_{K-\theta+3}(n)V_{K-\theta+4,\theta-2}(n) \\
& + \cdots + J_{N-1}(n)V_{N,\theta-2}(n)\} \\
& \vdots \\
& +U_{K-N+1}(n)J_{N-1}(n)V_{N,K-N+1}(n).
\end{aligned} \tag{D.5}
$$

When $q = K$, arriving high priority packets are always discarded; when $K - H + 1 \leq q \leq K - 1$, arriving high priority packets are discarded only if the number of arrivals from high priority streams exceeds the available buffer space; and when $K - N + 1 \leq q < \theta$, arriving packets are discarded randomly when the number of arrivals exceeds the available buffer space.

For the case when $\theta > K - H + 1$,

$$
\begin{aligned}
U(n) \;=\; & U_K(n) \\
& +U_{K-1}(n)\{J_1^h(n)V_{2,K-1}(n) + J_2^h(n)V_{3,K-1}(n) + \cdots + J_{H-1}^h(n)V_{H,K-1}(n)\} \\
& +U_{K-2}(n)\{J_2^h(n)V_{3,K-2}(n) + J_3^h(n)V_{4,K-2}(n) + \cdots + J_{H-1}^h(n)V_{H,K-2}(n)\} \\
& \vdots
\end{aligned}
$$

$$\begin{aligned}
&+U_{\theta}(n)\{J^{h}_{K-\theta}(n)V_{K-\theta+1,\theta}(n)+J^{h}_{K-\theta+1}(n)V_{K-\theta+2,\theta}(n)\\
&+\cdots+J^{h}_{H-1}(n)V_{H,\theta}(n)\}\\
&+U_{\theta-1}(n)\{J_{K-\theta+1}(n)V_{K-\theta+2,\theta-1}(n)+J_{K-\theta+2}(n)V_{K-\theta+3,\theta-1}(n)\\
&+\cdots+J_{N-1}(n)V_{N,\theta-1}(n)\}\\
&+U_{\theta-2}(n)\{J_{K-\theta+2}(n)V_{K-\theta+3,\theta-2}(n)+J_{K-\theta+3}(n)V_{K-\theta+4,\theta-2}(n)\\
&+\cdots+J_{N-1}(n)V_{N,\theta-2}(n)\}\\
&\vdots\\
&+U_{K-N+1}(n)J_{N-1}(n)V_{N,K-N+1}(n).
\end{aligned}\tag{D.6}$$

When $q = K$, arriving high priority packets are always discarded; when $\theta \leq q \leq K-1$, arriving high priority packets are discarded only if the number of arrivals from high priority streams exceeds the available buffer space; and when $K-N+1 \leq q < \theta$, arriving packets are discarded randomly when the number of arrivals exceeds the available buffer space. $P[V_{l,q}(n) = 1]$ is given in Eq.(D.3). For the remaining derivation, the same analytical technique used in section 4.2.2 can apply.