# Analyses of Heterogeneous Tandem Links, Part I: Per-Session Performance of an ATM Multiplexer

R. I. Balay
A. A. Nilsson

Center for Communications and Signal Processing
Department of Electrical and Computer Engineering
North Carolina State University

TR-94/20
October 1994

# Analyses of heterogeneous tandem links, part I: Per-session performance of an ATM multiplexer

**Rajesh I. Balay**
Department of Computer Science, and
Center for Communications and Signal Processing
North Carolina State University
Raleigh, N.C. 27695-7914

**Arne A. Nilsson**
Department of Electrical and Computer Engineering, and
Center for Communications and Signal Processing
North Carolina State University
Raleigh, N.C. 27695-7914

October 3, 1994

**Abstract**

In this paper we present analyses to study the performance of a session at a single switching node on an ATM network with heterogeneous speed links.

The traffic generated by an application (V-stream) traverses a set of links and switches on the network. Its performance is influenced by (1) the switching nodes which we assume to be non-blocking and have separate output buffers for each output link, (2) the links across the switch which may be of different speeds and use SONET/SDH framing protocol for transmitting ATM cells, and (3) the interfering (cross) traffic (X-stream) due to other sessions which use the same output link at a switching node. Since most applications generate bursty and correlated traffic we model V-stream as a Markov Modulated Bernoulli Process (MMBP) and X-stream as an MMBP with batch arrivals. The switching node is modeled as an $MMBP + MMBP^{[x]}/D(N_S)/1/K$ queueing system. We propose a novel method to obtain the exact steady-state performance results for V-stream and X-stream individually. We also present a very efficient approximation technique which reduces the complexity of our method while providing almost exact results.
*Keywords:* ATM; admission control; per-session performance; discrete-time modeling; finite capacity queue; Markov Modulated Bernoulli Process; bursty traffic; correlated arrivals; heterogeneous speed links; exact analysis.

# 1 Introduction

It is anticipated that Broadband Integrated Services Digital Networks (B-ISDNs) will support data, voice, video and multimedia applications using Asynchronous Transfer Mode (ATM) technology. One of the main aspects of ATM is statistical multiplexing of source traffic in the network to efficiently utilize the available bandwidth. This results in randomly varying delays and packet loss due to queueing at the switches. Most applications on these networks will have stringent quality of service (QoS) requirements on packet delay, delay variation and cell loss probability. An application when requesting a session specifies to the network the characteristics of its traffic and the performance it desires. Before granting a request for a new session the network makes sure that the requested QoS can be provided while ensuring that the QoS guarantees of existing sessions are not violated on accepting this new session. This task of admission control involves efficiently estimating (based on traffic descriptors and network status) the performance bounds for the new and existing sessions on the network. Also, for a network designer and a service provider it is often important to know exactly the performance of a session when the traffic characteristics and network topology are known. In this paper we address these issues by providing algorithmic solutions using discrete-time queueing models to obtain the exact performance of a session at a single switching node in a network with heterogeneous speed links.

Discrete-time queueing models have received considerable attention in communications modeling due to the increasing focus on issues related to ATM networks where packets are transmitted as fixed size units. The traffic on these networks is typically bursty and highly correlated ([15, 3, 14]). Complex, non-renewal processes are now needed to characterize these traffic streams, in contrast to the Poisson process which has traditionally been used to model teletraffic. The Markov Modulated Bernoulli Process (MMBP) ([12]), a non-renewal process which quantitatively models the time-varying behavior of arrival rates, and MMPP (Markov Modulated Poisson Process ([2]), the continuous time analogue of MMBP) have been used extensively to model traffic in communication networks. These processes capture the burstiness of a source and also some of the important correlations between the interarrival times, while still remaining analytically tractable.

Previous related work on discrete-time modeling for per-session analysis was mostly for renewal processes. Murata et al [8] analyzed a discrete-time system $GI + M^{[X]}/D/1/K$ for a single session performance where the session was modeled as a renewal process. In [9] Ohba and others extended such a system to provide end-to-end analysis of a stream modeled as a renewal process (GI-stream). Herrmann [4] considered non-renewal arrival processes for per-stream performance in a superposition by modeling a multiplexer as a $DMAP + DMAP/D/1/K$ queue. We consider a similar queueing system ($MMBP + MMBP^{[X]}/ D(N_S) / 1 / K$ queue) but use an alternate approach which is similar to that used in [8] and [9]. Here $D(N_S)$ denotes deterministic service time of $N$ slots per packet.

This paper is organized as follows: In section 2, we describe a switching scenario and define the analytical model and the assumptions made. In section 3, by exploiting the Markov renewal property of an MMBP, we present an exact analysis using a method which we call a 'system evolution approach', to derive the steady state distribution of system state (defined later) in a V-stream arrival slot. We use this distribution to obtain performance measures of V-stream and X-stream in sections 4 and 5. Section 6 describes an approximation which greatly reduces the computational cost of our method while maintaining the accuracy of the results. Later we present numerical results for V-stream as well as for X-stream in section 7. We make our concluding remarks in section 8.

# 2 ATM network scenario and analytical model

In this setion we discuss a network scenario which is of interest and develop a queueing model to analyze it. The focus of our research is to provide insight on the performance of a single application which makes use of broadband switching services. In the sub-sections below we briefly describe the network and its elements and their role in defining the queueing model.

## 2.1 ATM networks

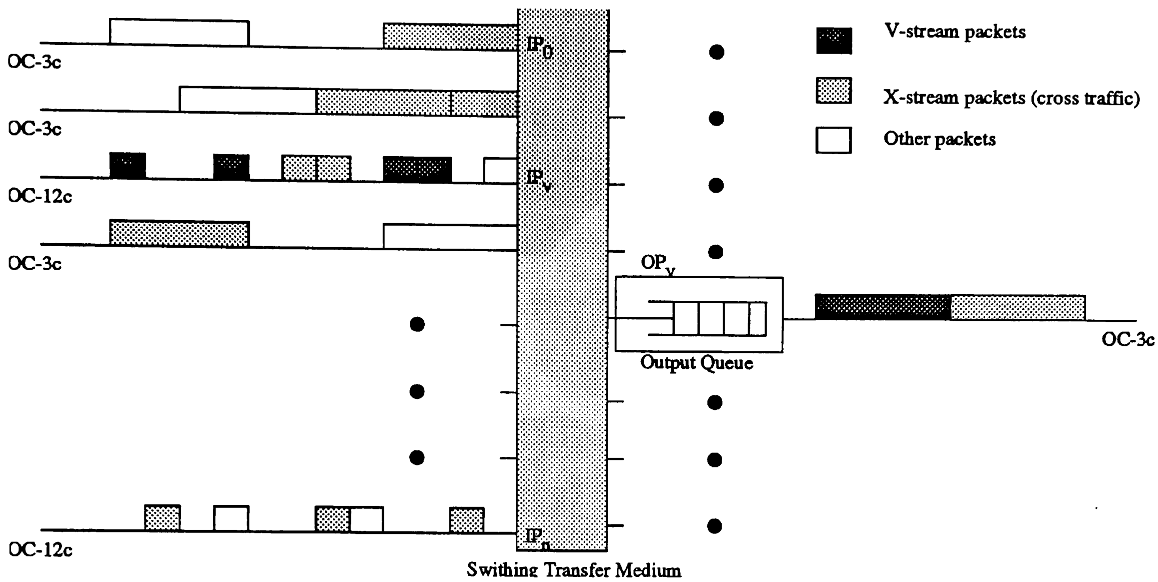
In ATM networks the transmission units are of fixed size of 53 bytes; we refer to them as packets in this paper. The transmission of packets across a network is based on the concept of virtual channels and virtual paths [11]. The network sets up a session for an application using virtual channel/path identifiers (VCIs/VPIs). Each cell header carries this information and is used for routing. The cells associated with an individual VCI/VPI are transported along the same route in the network. The route has a set of switches connected by links of different speeds in tandem. Since packets from differents sessions can be multiplexed onto a single link at a switch, the need for buffer arises. This introduces delay and also packet loss due to finite capacity of buffers. Different switch architecture have been proposed and developed to provide the performance quarantees of applications while optimizing the cost. In the next section we describe the ATM switch we consider for modeling.

### 2.1.1 Network model

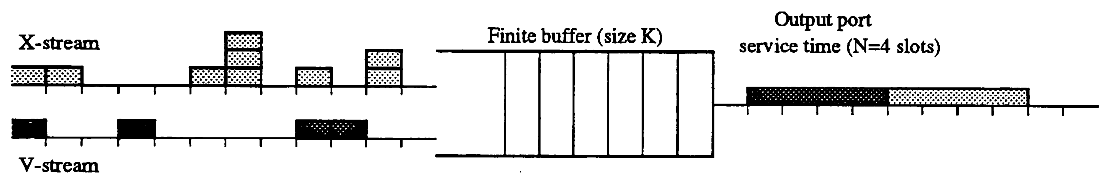
Due to the discrete time nature of arrivals and switching of ATM packets, we model an ATM network as a discrete time tandem queueing network; the nodes correspond to the switches present on the route of a session. We model each of these nodes separately and later combine the results to obtain approximate end-to-end performance measures..

## 2.2 ATM switch

The main function of a switch is to transfer incoming ATM packets to their respective output links based on the VCI/VPI routing information. A switch may consist of a single switching element or a fabric of switching elements, each consisting of input ports (IPs), output ports (OPs) and switching transfer medium for transferring packets from the IPs to the respective OP. Packets arrive at an IP from a framer chip after they are recovered from an arriving SONET frame, demultiplexed (if a multiplexed SONET signal was used) and are synchronized to the internal clock of the switch. A framer chip at each output




(a) A switching element with output buffers




(b) Queueing model of switching element

Figure 1: An example switching scenario and the corresponding queueing model

link either concatenates packets at a single OP or multiplexes packets from many OPs into SONET frames. We consider our model switch to consist of a single switching element with a separate buffer at each OP. The switching transfer medium can be realized by a high speed time division multiplexing (TDM) bus. The speed of the bus is at least the sum of capacities of all input links. This avoids contention by the IPs for transmission and each packet arriving at an input port in a slot is transmitted immediately over the bus using a bus access algorithm. Multiple packets directed to the same output port are buffered; a packet is lost if it arrives whe the buffer is full. We consider the bus access algorithm to service input ports sequentially from port $IP_1$ to port $IP_n$ in each slot and in the same order every slot. This means that the performance of a session depends on the arrival port at a switch since packets from a session are more likely (or less likely) to get lost if they arrive at the

OP buffer after (rather than before) the packets from other sessions in the same slot.

### 2.2.1 Switch model

Packets from a session of interest (V-stream) arrive at a particular IP (say $IP_V$) and are routes to an OP ($OP_V$). The packets belonging to all other sessions which are routed to $OP_V$ constitute the cross-traffic (X-stream). We model $OP_V$ of our model switch as a discrete-time, finite buffer capacity, single server queue. The packets from the two streams arrive at a common buffer of size $K$ (packets). A packet is discarded if it arrives when the buffer is full. If packets arrive from both the streams in the same, the we consider one of the two policies stated below:

- Policy $\mathcal{P}_V$ : V-stream arrival in a slot is considered first for buffer allocation. This case corresponds to having V-stream packets arrive at the first input port at a switch ($IP_1$).
- Policy $\mathcal{P}_X$ : X-stream packets arriving in a slot are considered for buffer allocation before before the X-stream packet in the same slot. This case corresponds to having V-stream packet arrive at the last input port at a switch ($IP_n$).

These two policies provide us with bounds on performance variations due to location of arrival port at a switch.

We consider 'early arrivals' where the arrival events occur before the departure event in the same slot. A packet arriving in a slot cannot receive service in the same slot i.e., a packet arriving in a slot which finds the server to be idle will receive service only in the next slot. Each packet takes $N_S$ slots (described below) of service time and is processed in the same order of its placement in the FIFO queue.

## 2.3 SONET/SDH

SONET (synchronous optical network) provides a vehicle for synchronously multiplexing a variety of lower speed signals using a standardized hierarchy of transmission rates. The basic modular signal for the SONET hierarchy is STS-1 (synchronous transport level signal-1) with transmission bit rate of 51.84 Mb/s. An higher level signal STS-N can be formed by byte interleaving N STS-1 signals. Applications such as high definition TV (HDTV) require transmission capacity of 600 Mb/s. To accommodate such services which require greater transmission capacity than STS-1, SONET defines 'concatenated' hierarchical signals. A concatenated STS-N signal is denoted by STS-Nc and is created directly rather than by multiplexing basic rate tributaries. A SONET frame ([10]) corresponding to STS-N or STS-Nc signal can be transmitted at a rate denoted by OC-N (optical carrier-N). Transmission rates standardized so far by CCITT are OC-1 at 51.84Mb/s, OC-3 at 155.52 Mb/s, OC-12 at 622 Mb/s and OC-48 at 2.49 Gb/s. We will refer to transmission rates of an STS-Nc frame as OC-Nc rather than OC-N to distinguish it from the STS-N signal transmission. We consider STS-3c framing as the lowest level used for encapsulating ATM packets. ATM packets are framed and recovered at the switch interfaces and at edges of the networks i.e., at the user network interfaces (UNIs). A few prototype integrated circuits (ICs) which perform the functions defined by SONET/ATM standards are discussed in [7, 13, 5].

### 2.3.1 Slotted system model

The deterministic service time of our queueing model depends on the SONET framing protocols used at the $IP_V$ and $OP_V$ which may be different. To define $N_S$ we define a *link slot* as the time to transmit a packet on a link, and a *model slot* (the time slot used for the queueing model) as the greatest common factor of the input and output link slots. Consider a case where the input and output links are of speeds OC-12c and OC-3c respectively. Figure 1 shows a switching element and the corresponding queueing model for such a scenario. Here, a packet arrives four times faster than it can be transmitted by $OP_V$, i.e., a packet can arrive in every model slot but each packet takes for model slots for transmission. The deterministic service time ($N_S$) of our queueing model is the packet transmission time in model slots, i.e., for this example $N_S = 4$. If we have input link speed at $IP_V$ as OC-3c and the sped at $OP_V$ as OC-12c then each packet takes four model slots to arrive and one model slot for transmission. We define $N_A$ as the models slots for a V-stream arrival. We assume that X-stream packet arrivals always take one model slot to arrive since the packets which constitute X-stream arrive from different speed links. The model parameters for a few cases

| Input Link ($IP_V$) | Output Link ($OP_V$) | Arrival slots ($N_A$) | Service slots ($N_S$) |
|---|---|---|---|
| OC-3c | OC-3c | 1 | 1 |
| OC-12c | OC-3c | 1 | 4 |
| OC-12 | OC-3c | 1 | 1 |
| OC-12 | OC-12c | 4 | 1 |
| OC-3c | OC-12c | 4 | 1 |
| OC-48 | OC-3c | 1<br>1 | 1<br>4 |

Table 1: Link speeds and corresponding model parameters

of link speeds are given in table 1. Notice than when we have an OC-12 link we consider it same as an OC-3c for modeling since it has four multiplexed streams and the rate at which V-stream packets are received is OC-3c. System where $N_A > 1$ are analyzed by considering an arrival to take one model slot and having arrivals occur only at multiples of $N_A$ slots apart (by modifying the arrival process). From now onwards we refer to a model slot as a slot in our paper.

## 2.4 ATM applications and traffic characteristics

ATM networks will support all the present communication services such as data, voice, video and many new applications such as video on demand, video conferencing and multimedia. Most applications will have stringent performance requirements in terms of packet loss and delay. The traffic generated by such applications has been measured and studied for its characteristics. It has been found that the traffic is highly bursty in nature and the interarrival times are correlated. These characteristic change when a traffic stream passes from one node to the other within a network.

### 2.4.1 Traffic model

The V-stream is modeled as a Markov Modulated Bernoulli Process (MMBP) which captures the notion of burstiness and the correlation properties of an arrival stream. An MMBP is a Bernoulli process where the arrival rate is varied according to an m-state Markov chain. This variation using state changes in an MMBP helps capture the notion of burstiness and correlation properties of the arrival stream. An MMBP is characterized by the state-transition probability matrix $P_t$ of the Markov chain and the arrival rate descriptor $\Lambda$ defined as follows:

$$\mathbf{P_t} = \begin{bmatrix} p_{11} & & p_{1m} \\ & \ddots & \\ p_{m1} & & p_{mm} \end{bmatrix} \quad \text{and} \quad \mathbf{\Lambda} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{bmatrix},$$

where $p_{ij}, 1 \leq i, j \leq m$, is the transition probability that the state changes from $i$ to $j$ and $\sum_{j=1}^{m} p_{ij} = 1$ for $1 \leq i \leq m$. $\alpha_i$ is the probability of having an arrival in a slot when the MMBP is in state $i$, $1 \leq i \leq m$.

A traffic stream is characterized by three descriptors: mean packet arrival rate ($\lambda$), burstiness defined as the squared coefficient of variation ($C^2$) and autocorrelation of lag 1 slot ($\psi_1$). Given certain values of offered load, burstiness and autocorrelation we can obtain the inter-arrival time distribution of the V-stream arrival process conditioned on the MMBP state of the previous arrival (used in this form in our derivations) which is given as
$a_v(v', j)$ = P[*next V-stream arrival occurs in MMBP state* $v'$ *and after* $j$ *slots* | *a V-stream arrival occurred when MMBP state was* $v$].
The derivation of $a_v(v', j)$ for a 2-state MMBP from traffic descriptors is given in Appendix A.

The X-stream is modeled as a Markov Modulated Bernoulli Process with batch arrivals ($MMBP^{[x]}$). An $m$ state Markov chain is used for the MMBP as described above for V-stream traffic. Each arrival brings into the system a batch of packets. The batch size (no. of packets in an arrival) follow a general distribution. Given that the MMBP is in a particular state $x$, the probability density of the number of packets arriving in that slot $b_x(j)$ = P[*X-stream packets arriving in a slot* $= j \mid$ *MMBP state* $= x$], $j \geq 0$, is given as

$$b_x(j) \quad = \quad \begin{cases} 1 - \alpha_x & if \quad j = 0 \\ \alpha_x s(j) & otherwise \end{cases} \tag{1}$$

where $\alpha_x$ is the probability of having an arrival when the X-stream MMBP is in state $x$ and $s(j), j > 0$ is the probability density that the batch size is $j$. For a geometric distribution the density function is given as $s(j) = (1-x)^{j-1}x$ where $j > 0$, $0 < x < 1$ and $x = \frac{1}{Mean\ Batch\ Size}$.

# 3 Exact analysis using system evolution approach

To obtain steady-state performance measures of the system we use a method similar to 'method of embedded points' described by Kleinrock ([6]). We consider the V-stream packet arrival instants as the embedded points in our system. For a simple system ($N_A = 1$,$N_S = 1$ and Bernoulli arrival processes) let number of packets in the system be defined as its state; a high level description of system evolution approach for this system is:

(1) start with an arbitrary initial state at an embedded point

(2) *evolution step*: obtain the state distribution at the next embedded point using the current state information and the arrival characteristics

(3) repeat the above evolution step until the state distribution at two consecutive embedded points is the same, i.e., evolve the system until it reaches steady-state.

If V-stream arrivals are renewal with arrival epochs ($T_n$), then the interarrival times ($I \in \{I_n : I_n = T_n - T_{n-1}, n \in \mathcal{N}\}$) are identically and independently distributed (i.i.d) and the V-stream arrival instants are regeneration points (the elapsed time since last arrival regenerates). Let $C_k^{(n)}$ denote the system state in the $k^{th}$ slot following a regeneration point

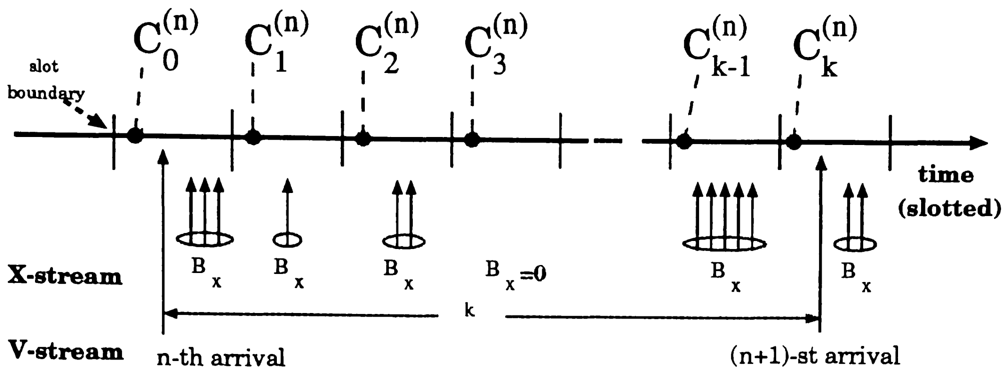



Figure 2: Relating system states observed by two consecutive V-stream arrivals

and in the $n^{th}$ evolution step, and $f_I$ be the p.d.f. of interarrival time for V-stream. The evolution step for this system is:

$$c_0^{(n+1)}(j) = \sum_{k=1}^{\infty} f_I(k)\, c_k^{(n)}(j).$$

For the case where V-stream is modeled as an MMBP we make use of the Markov renewal property of the MMBP ([1, 2]). Note that an MMBP is not a *renewal process* but a *Markov renewal process*: the distribution of the time between $(n-1)^{st}$ arrival and $n^{th}$ arrival depends on the MMBP state at the times of the $(n-1)^{st}$ and $n^{th}$ arrivals. The sequence $(V,T) = \{(V_n, T_n); n \in \mathcal{N}\}$, is a Markov renewal process, where $V$ is the MMBP state and $T$ is the arrival epoch. The interarrival time $I_{v,v'} \in \{I_{v,v',n} : I_{v,v',n} = T_n - T_{n-1} \mid V_{n-1} = v, V_n = v', n \in \mathcal{N}\}$, clearly represents a renewal process. Hence, to apply system evolution approach we use conditional interarrival time p.d.f. (see section (2.4.1)) for V-stream and we also need to redefine our system state.

## 3.1 Steady state distribution of system state in a V-stream arrival slot

We define $C_k^{(n)}$ as the random variable used to represent the system state at the beginning of the $k^{th}$ slot following the $n^{th}$ V-stream arrival (i.e. in the $n^{th}$ evolution step) given that the $(n+1)^{st}$ V-stream arrival does not occur in the preceding $k-1$ slots (see figure 2). $C_k^{(n)}$ is a 4-tuple, $C_k^{(n)} \equiv (V_k^{(n)}, X_k^{(n)}, S_k^{(n)}, J_k^{(n)})$ (with its p.d.f. defined as $c_k^{(n)}(v,x,s,j) = P[V_k^{(n)} = v, X_k^{(n)} = x,\ S_k^{(n)} = s, J_k^{(n)} = j])$, $k \geq 0, n \geq 1$, where

$V_k^{(n)}$ : state of V-stream MMBP (V-MMBP) in the $n^{th}$ V-stream arrival slot. Note that this variable does not change with $k$, i.e. $V_k^{(n)} = V_0^{(n)}$, $\forall k > 0$. The state space of $V_k^{(n)}$ is represented as $\mathcal{V}$.

$X_k^{(n)}$ : state of X-stream MMBP (X-MMBP) in a slot. The state space of $X_k^{(n)}$ is represented as $\mathcal{X}$.

$S_k^{(n)}$ : state of the server. The server is either in an idle state ($S_k^{(n)} = 0$) or is in a busy state given by a positive value $S_k^{(n)} = i$ ($\leq N_S$), which indicates that the current slot is the $i^{th}$ service slot for the packet in service. A departure occurs at the end of slot $k$ if $S_k^{(n)} = N_S$.

$J_k^{(n)}$ : number of packets in system (i.e. queue and server).

We consider $C_0^{(n)}$ as the system state at an embedded point; a deviation from our previous definition since it is not necessarily the state at an arrival instant. In terms of regeneration this makes no difference since arrivals are slotted. The advantage is that from steady-state distribution of $C_0^{(n)}$, we can obtain performance measures of both streams and for both policies ($\mathcal{P}_V$ and $\mathcal{P}_X$). Our objective is to obtain the system state at the next embedded point (V-stream arrival slot). For which we need to obtain system state at each consecutive slot following a V-stream arrival (up to the next arrival slot). The events which influence the change between system states at two consecutive slots (slot $k-1$ and slot $k$) are

($i$) the V-stream packet arrival in slot $k-1$ if $k=1$,

($ii$) a batch arrival of packets from the X-stream in slot $k-1$,

($iii$) a departure from the system at the end of the slot $k-1$ and

($iv$) the server state change at the beginning of slot $k$,

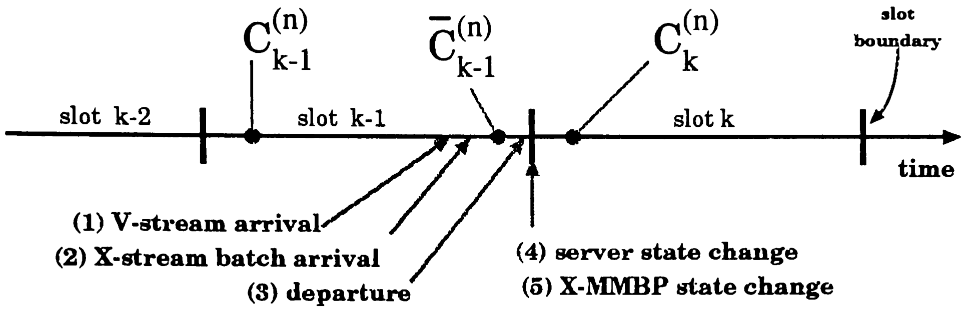



Figure 3: Relative order of events and observations in a slot (policy $\mathcal{P}_V$)

($v$) X-MMBP state change at the beginning of slot $k$.

For ease of derivation we consider the change in system state due to these events in two phases introducing an intermediate system state $\bar{C}_{k-1}^{(n)}$ (as shown in figure 3).

**phase (1)**: the arrivals which occur in slot $k-1$ (events (i) and (ii))

**phase (2)**: the state transitions at the slot boundary between slots $k-1$ and $k$ and the departure (if $S_{k-1}^{(n)} = N_S$) at the end of slot $k-1$ (events (iii),(iv) and (v)).

$\bar{C}_{k-1}^{(n)} = (\bar{V}_{k-1}^{(n)}, \bar{X}_{k-1}^{(n)}, \bar{S}_{k-1}^{(n)}, \bar{J}_{k-1}^{(n)})$ (p.d.f. represented as $c_{k-1}^{(n)}(\cdot)$), is the system state after **phase (1)**, where $\bar{V}_{k-1}^{(n)} = V_{k-1}^{(n)}$, $\bar{X}_{k-1}^{(n)} = X_{k-1}^{(n)}$, $\bar{S}_{k-1}^{(n)} = S_{k-1}^{(n)}$, and

$$\bar{J}_{k-1}^{(n)} = min(J_0^{(n)} + B_x + \bar{u}(k-1), K + u(S_{k-1}^{(n)})) \tag{2}$$

where $k \geq 1$, $n > 0$, $B_x$ is the conditional r.v. for X-stream batch size (p.d.f. defined in (1)), $\bar{u}(n) \stackrel{\text{def}}{=} 1 - u(n)$, and

$$u(n) = \begin{cases} 0 & if\ n = 0 \\ 1 & if\ n > 0. \end{cases}$$

Equation (2) accounts for the facts that a V-stream arrival needs to be considered only in slot *0*, and the maximum packets that can be accommodated depends on system state ( i.e., $K$ if the system is initially idle or else $K+1$). The expressions to derive $C_k^{(n)}$ from $\bar{C}_{k-1}^{(n)}$ (corresponding to **phase 2**) are given below.

$$V_k^{(n)} = \bar{V}_{k-1}^{(n)}, \quad J_k^{(n)} = \bar{J}_{k-1}^{(n)} - u(N_S - S_{k-1}^{(n)}),$$

$$X_k^{(n)} = x \text{ with probability } p_{i,x} \text{ where } \bar{X}_{k-1}^n = i, \ x \in \mathcal{X} \tag{3}$$

$$S_k^{(n)} = \begin{cases} \bar{S}_{k-1}^n + 1 & if \quad 0 < \bar{S}_{k-1}^n < N_S \\ 1 & if \quad \bar{S}_{k-1}^n = 0, \ \bar{J}_{k-1}^{(n)} > 1 \quad or \quad \bar{S}_{k-1}^n = N_S, \ \bar{J}_{k-1}^{(n)} > 0 \\ 0 & if \quad \bar{S}_{k-1}^n = 0, \ \bar{J}_{k-1}^{(n)} = 1 \quad or \quad \bar{S}_{k-1}^n = N_S, \ \bar{J}_{k-1}^{(n)} = 0 \end{cases} \tag{4}$$

where $k \geq 1$ and $n \geq 1$. Let $\tilde{n}_k \stackrel{\text{def}}{=} K + u(S_k) - \bar{u}(k)$. The relation between the p.d.f.s corresponding to equation (2) (**phase(1)**) is

$$\bar{c}_k^{(n)}(v, x, s, j) = \begin{cases} \sum_{i=0}^{j-\bar{u}(k)} c_k^{(n)}(v, x, s, i) \, b_x(j - i - \bar{u}(k)) & if \quad j < \tilde{n}_k \\ \sum_{l=\tilde{n}_k}^{\infty} \sum_{i=0}^{j-\bar{u}(k)} c_k^{(n)}(v, x, s, i) \, b_x(l - i - \bar{u}(k)) & if \quad j = \tilde{n}_k \end{cases} \tag{5}$$

The equations corresponding to **phase(2)** where state transitions and a possible departure can occur are

$$c_k^{(n)}(v, x, s, j) = \begin{cases} \sum_{x'} p_{x',x}(\bar{c}_{k-1}^{(n)}(v, x', 0, 0) + \bar{c}_{k-1}^{(n)}(v, x', N_S, 1)) \\ \qquad if \quad s = 0, \ j = 0 \\ \sum_{x'} p_{x',x}(\bar{c}_{k-1}^{(n)}(v, x', 0, j) + \bar{c}_{k-1}^{(n)}(v, x', N_S, j+1)) \\ \qquad if \quad s = 1, \ 1 \leq j \leq K \\ \sum_{x'} p_{x',x}\bar{c}_{k-1}^{(n)}(v, x', s-1, j+1) \\ \qquad if \quad 2 \leq s \leq N_S, \ 1 \leq j \leq K+1 \\ 0 \qquad otherwise \end{cases} \tag{6}$$

where $k \geq 1$, $n \geq 1$, $x, x' \in \mathcal{X}$, $v \in \mathcal{V}$ and $p_{x',x}$ is the probability of transition from state $x'$ to state $x$ for X-MMBP.

Starting with an arbitrary distribution of system state at an embedded point $(c_0^{(n=1)}(v, x, s, j))$, we can apply equations (5) to (6) corresponding to **phase(1)** and **phase(2)** repeatedly to obtain the system state distribution at subsequent slots $(c_k^{(n)}(v, x, s, j), for \ \ k = 1, 2, \cdots)$. The evolution step to obtain the p.d.f. of system state at the next embedded point ($(n+1)^{st}$ V-stream arrival slot) is given as

$$c_0^{(n+1)}(v', x, s, j) = \sum_k \sum_v a_v(v', k) \, c_k^{(n)}(v, x, s, j), \tag{7}$$

where $k \geq 1$, $v, v' \in \mathcal{V}$, $x \in \mathcal{X}$, $0 \leq s \leq N_S$ *and* $0 \leq j \leq K+1$.

If steady state exists then the steady state p.d.f. of the system state at an embedded point (i.e. a V-stream arrival slot) is given as

$$c_0(v,x,s,j) = \lim_{n \to \infty} c_0^{(n)}(v,x,s,j). \tag{8}$$

Using the system state p.d.f. $c_0(v,x,s,j)$ we can now obtain the density functions of the system state seen by V-stream arrivals and X-stream arrivals, see section 4 and 5. Let the p.d.f. of system state seen by a particular stream be $c(\cdot)$. Some of the interesting performance measures are found as follows:

p.d.f. of buffer occupancy

Let $f_Q(j)$ represent the p.d.f. of buffer occupancy (queue length) seen by the arriving packets. It is given as

$$f_Q(j) \quad = \sum_{v,x} \left[ c(v,x,0,j) \;+\; \sum_{s=1}^{n} c(v,x,s,j+1) \right] \quad \text{for } 0 \leq j \leq K. \tag{9}$$

packet loss probability

The probability of loss ($P_{loss} = \mathrm{P}[\textit{packet is lost}]$) is

$$P_{loss} = \sum_{v,x} \left[ c(v,x,0,K) \;+\; \sum_{s=1}^{N_S} c(v,x,s,K+1) \right]. \tag{10}$$

delay distribution

The delay in the system is defined as the time spent by a packet in system excluding the arrival slot and including the departure slot. The p.d.f. of the delay in system, $f_D(j) = \mathrm{P}[\textit{packet delay} = j \textit{ slots}]$, is given as

$$f_D(j) \quad = \quad \begin{cases} \sum_{v,x} c(v,x,0,l-1) \;+\; \sum_{v,x} c(v,x,N_S,l) & \text{if } j \bmod N_S = 0 \\ \sum_{v,x} c(v,x,s,l) \quad \text{if } j \bmod N_S \neq 0 \text{ and } N_S \leq j < (K+1)N_S \\ 0 \quad \text{otherwise} \end{cases} \tag{11}$$

where $s = (N_S - j) \bmod (N_S)$ and $l = \lfloor j/N_S \rfloor$, i.e., $s$ and $l$ satisfy $j = N_S - s + lN_S$, where the term $(N_S - s)$ represents delay caused by a packet in service at the V-stream arrival instant and the term $(lN_S)$ represents delay due to service time of buffered packets and the current packet.

# 4 Performance of a single session (V-stream)

We are interested in finding the system state seen by arrivals from V-stream since we can apply equations (10) and (11) to obtain the performance results. Let $C^v = (V^v, X^v, S^v, J^v)$ be the random variable for the system state seen by a V-stream arrival at steady state and let $c^v(\cdot)$ be its p.d.f.

**Case 1: $\mathcal{P}_V$ arrival policy**

This is the case where a V-stream arrival is considered before the packets from the X-stream for buffer allocation. The system state $(C^v)$ seen by a V-stream arrival is the same as that at the beginning of the slot, i.e., $C^v \equiv C_0$. We have lower bounds on V-stream loss probabilities (not necessarily lower bounds on delay, explained in numerical results) when using policy $\mathcal{P}_V$.

**Case 2: $\mathcal{P}_X$ arrival policy**

Here the X-stream packets arriving in a slot have priority over the V-stream arrival in the same slot. By considering the X-stream batch arrival ( $B_x$) in a V-stream arrival slot we obtain $C^v$ from $C_0$. The relations between the r.v's and p.d.f's are

$$V^v = V_0, \quad X^v = X_0, \quad S^v = S_0, \quad J^v = min(J_0 + B_x, K + u(S_0)). \tag{12}$$

$$c^v(v,x,s,j) = \begin{cases} \sum_{i=0}^{j} c(v,x,s,i)\; b_x(j-i) & if \quad j \; < \; K + u(s) \\ \sum_{l=K+u(s)}^{\infty} \; \sum_{i=0}^{j} c(v,x,s,i)\; b_x(j-i) & if \quad j \; = \; K + u(s). \end{cases} \tag{13}$$

# 5 Performance of cross traffic (X-stream)

Consider all X-stream packets which arrive between two consecutive V-stream arrivals in steady-state. Let a packet from this set be called 'test packet'. The expected p.d.f. of system seen by the test packet represents the p.d.f. ($c^x(\cdot)$) of system state ($C^x$)seen by any X-stream packet. The position of the test packet is given by two independent r.v's $H$ and $L$, where $H$ is the position of the test packet in the batch and $L$ is the arrival slot of the batch. Using the p.d.f's of $C_0$ (system state at the beginning of a V-stream arrival slot), $H$ and $L$ we derive the p.d.f. of system state seen by an X-stream arrival in section 5.3. First we derive the p.d.f. of $H$ and $L$, respectively.

## 5.1 Arrival slot of an X-stream packet ($L$)

Let $s_v(l)$ be the conditional p.d.f. of $L$ given that the previous V-stream arrival occurred when MMBP state was $v$. If the policy is $\mathcal{P}_v$ and if the next V-stream arrival occurs in slot $k$ then the test packet could arrive in either of slots $\{0, 1, \cdots, \ k-1\}$. We derive $s_v(l)$ as follows:

$$\begin{aligned} P[Next\ V\ stream\ arvl\ in\ k^{th} slot \mid V_0 = v] &= \sum_{v'} a_v(v', k) \\ P[Next\ V\ stream\ arvl\ in\ k^{th} slot\ and\ (0,k)\ has\ test\ pkt \mid V_0 = v] &= k \sum_{v'} a_v(v', k)/G_v \end{aligned}$$

where $G_v$ is the expected slot of arrival of a V-stream packet given the previous V-stream arrival occurred in MMBP state $v$, i.e.,

$$G_v = \sum_{i=1}^{\infty} \sum_{v'} i\ a_v(v', i)$$

Conditional p.d.f. of the test packet arrival slot is given as

$$\begin{aligned} s_v(l) &= P[L = l \mid V_0 = v] = \sum_{k=l+1}^{\infty} \frac{1}{k}\, k \sum_{v'} a_v(v', k)/G_v \\ &= \frac{1}{G_v} \sum_{k=l+1}^{\infty} \sum_{v'} a_v(v', k) \end{aligned} \tag{14}$$

where $l \geq 0$ and $v, v' \in \mathcal{V}$. If the policy is $\mathcal{P}_x$ then the domain of the test packet arrival slot is $\{1, \cdots, k\}$ where $k$ is the next V-stream arrival slot. We can obtain $s_v(l)$ as above and is given as

$$s_v(l) = \begin{cases} 0 & if \;\; l = 0 \\ \frac{1}{G_v} \sum_{k=l+1}^{\infty} \sum_{v'} a_v(v', k) & otherwise. \end{cases} \tag{15}$$

where $l \geq 0$ and $v, v' \in \mathcal{V}$.

## 5.2 Position of an X-stream packet in a batch ($H$)

Let $h_x(i)$ be the conditional p.d.f. of $H$ given the X-MMBP in the slot is $x$. To derive it we first define the following r.v's

(1) B : no. of packets in an X-stream batch arrival

(2) $B_x$ : no. of packets in an X-stream batch arrival given X-MMBP is '$x$'

(3) $B''_x$ : no. of packets in an X-stream batch arrival given X-MMBP is '$x$'and at least one packet in the X-stream batch in that slot

(4) $B'_x$ : no. of packets ahead of the the test packet in an X-stream batch arrival given X-MMBP is '$x$' in that slot.

Also, we define the p.d.f of $B''_x$ and the conditional expected batch size $\bar{b}''_x$ as

$$\begin{aligned} b''_x(m) &= P[B''_x = m] = P[B = m \; and \; B > 0 \mid X = x] \\ &= \frac{b_x(m)}{1 - b_x(0)} \quad where \;\; x \in \mathcal{X} \;\; and \;\; m \geq 1, \\ \bar{b}''_x &= \sum_{m=1}^{\infty} m \; b''_x(m) \qquad x \in \mathcal{X}. \end{aligned}$$

We derive $h_x(i)$ as follows

$$\begin{aligned} h_x(i) &= P[H = i \mid X = x] = \sum_{m=i}^{\infty} P[B''_x = m \; and \; B''_x \; contains \; test \; packet] \\ &= \sum_{m=i}^{\infty} \frac{1}{m} \frac{m \; b''_x(m)}{\bar{b}''_x} = \frac{1}{\bar{b}''_x} \sum_{m=i}^{\infty} b''_x(m), \qquad i \geq 1. \end{aligned} \tag{16}$$

For deriving p.d.f. of system state seen an X-stream arrival we make use of of p.d.f. of $B'_x = H_x - 1$. Its p.d.f. is given as

$$b'_x(i) = \frac{1}{\overline{b''_x}} \sum_{m=i+1}^{\infty} b''_x(m), \qquad i \geq 0. \tag{17}$$

## 5.3 System state seen by an X-stream packet

To obtain the system state seen by any X-stream packet we first calculate the system state seen by a test packet given its position. We uncondition it using the distributions of position of a test packet derived in the previous two subsections. A systematic method to obtain the system state seen by any X-stream packet is to derive the following listed r.v's in that order from their previous ones.

(1) $C_0$ : system state at the beginning of a V-stream arrival slot,

(2) $C_k$ : system state at the beginning of $k^{th}$ slot following a V-stream arrival which occurred in slot 0,

(3) $C'_k$ : system state at the beginning of $k^{th}$ slot when $L = k$, i.e., given the test packet arrives in slot $k$ ($C'_k = C_k | L = k$),

(4) $C^x_k$ : system state seen by the test packet given that the test packet arrives in slot $k$ ($C^x_k = C^x | L = k$),

(5) $C^x$ : system state seen by the test packet, i.e., any X-stream packet.

We have previously derived the relation to obtain $C_k$ from $C_0$; below we derive the other r.v's and their p.d.f's.

### 5.3.1 System state at beginning of test packet arrival slot ($C'_k$)

The test packet arrival slot is distinctly different from any arbitrary slot since there is at least one packet from X-stream arriving in the slot. To reflect this we need to condition $X_k$

as an arrival state of X-MMBP. We introduce a new system state $C'_k$, and obtain it using $\pi_X(x)$ and $\tilde{\pi}_X(x)$ which denote the probability that X-MMBP in a slot at steady-state is $x$ and probability that X-MMBP state is $x$ in an arrival slot at steady-state, respectively.

$$\begin{aligned} c'_k(v,x,s,j) &= P[C'_k = (v,x,s,j)] = P[C_k = (v,x,s,j) \mid L = k] \\ &= P[C_k = (v,x,s,j) \mid X_k = x]\ P[X_k = x \mid L = k] \\ &= P[C_k = (v,x,s,j) \mid X_k = x]\ P[X_k = x \mid B > 0] \\ &= \frac{c_k(v,x,s,j)}{\pi_X(x)}\ \tilde{\pi}_X(x) \end{aligned} \tag{18}$$

### 5.3.2 System state seen by an X-stream packet in a test packet arrival slot ($C^x_k$)

Let the test packet arrival slot be $k$ (i.e. $L = k$). If the policy is $\mathcal{P}_V$ then $k \geq 0$ since we may have the test packet arrive in the V-stream arrival slot i.e., slot 0, else we have $k > 0$. We obtain the system state $C^x_k$ from the system state at the beginning of a test packet arrival slot ($C'_k$) by considering all the other X-stream packet in the batch which are ahead ($B'_x$) of this packet in the batch. Also the V-stream arrival need to be considered if $k = 0$. The relations between the r.v's $C^x_k$ and $C'_k$ are given as follows

$$V^x_k = V'_k, X^x_k = X'_k, S^x_k = S'_k, J^x_k = min(J'_k + B'_x + \bar{u}(k), K + u(S_k)) \tag{19}$$

$$c^x_k(v,x,s,j) = \begin{cases} \displaystyle\sum_{i=0}^{j-\bar{u}(k)} c'_k(v,x,s,i)\ b'_x(j-i-\bar{u}(k)) & if \quad j < \tilde{n}_k \\ \displaystyle\sum_{l=\tilde{n}_k}^{\infty} \sum_{i=0}^{j-\bar{u}(k)} c'_k(v,x,s,i)\ b'_x(j-i-\bar{u}(k)) & if \quad j = \tilde{n}_k \end{cases} \tag{20}$$

### 5.3.3 System state seen by an X-stream packet

We obtain the system state seen by any X-stream packet by unconditioning the system state $C^x_k$ on the test packet arrival slot using the p.d.f. $s_v(k)$. as

$$c^x(v,x,s,j) = P[C^x = (v,x,s,j)] = \sum_k P[C^x(v,x,s,j) \mid L = k]\ P[L = k \mid V_0 = v]$$

$$= \sum_k s_v(k)\; c_k^x(v,x,s,j) \tag{21}$$

Equations (9) to (11) are used to obtain the performance measures of X-stream.

# 6 An approximation technique for system evolution approach

The system evolution approach described in this paper provides us with exact performance results at a computational cost mainly in two dimensions:

(1) computing the iterative step i.e., computation of $c_k^{(n)}(\cdot)$, for $k \geq 0$.

(2) number of iterations for convergence of system state p.d.f.

In this section we address the issue of reducing computational cost in the first dimension i.e., the iterative step. The method described can also be used for reducing the computational cost of the second dimension.

## 6.1 Obtaining exact results

Let $M_A$ represent the maximum value for the interarrival time in slots for V-stream arrivals. In theory, the value of $M_A$ could be infinite but for practical solutions of exact analysis of any system we need finite values. By using a finite value for $M_A$ we get exact results when the probability density function of interarrival time for V-stream arrivals ($a(j)$) is completely specified with the chosen constant $M_A$ i.e., $\sum_{j=1}^{M_A} a(j) = 1$. Almost all distributions have infinite tail values, so we consider a tolerance $\delta$ for specifying the arrival p.d.f. with reasonable accuracy i.e., we chose a value for $M_A$ such that it is the least value which satisfies the inequality $1-\delta \leq \sum_{j=1}^{M_A} a(j) \leq 1$. For bursty processes and when the offered load of a process is low, the constant $M_A$ required for a given tolerance is large; hence a high computational cost.

### 6.1.1 Method of tail truncation

Sometimes exact numerical solutions with a given tolerance are infeasible due to computational constraints. To avoid such scenarios we choose a lower value for $M_A$ than required and modify the arrival process. By limiting $M_A$ we trade with the accuracy in specifying the arrival process. We modify the arrival process by truncating the p.d.f. of interarrival times $(a(j))$ at $j = M_A$, and add the truncated tail values $(a(j), j > M_A)$ to the truncated point, i.e.,

$$a(j) = \begin{cases} a(j) & if \;\; j < M_A \\ a(j) + \delta' & if \;\; j = M_A \\ 0 & otherwise \end{cases} \tag{22}$$

where $\delta' = \sum_{j=M_A+1}^{\infty} a(j)$. On modifying an arrival process as above, its characteristics (such as offered load, burstiness index) change significantly. This results in error in numerical solutions. This approximation is widely used due to its simplicity and as a first step in limiting the tail of an arrival process in numerical methods. It provides us with results when we have complex arrival processes but again at a considerable computational cost.

## 6.2 Geometric tail approximation of evolution step

We are interested in approximations which provide accurate results while they are efficiently implementable. To provide such approximations we often seek closed form solutions by exploiting or approximating the process or system characteristics. In this section we present such an approximation which proves to be very efficient. First, we describe the approximation for the case of homogeneous speed links and later extend it to the case of heterogeneous speed links.

### 6.2.1 Case 1: Homogeneous speed links

In each evolution step while computing $C_0^{(n+1)}$ from $C_0^{(n)}$, the system is initially (in slot 0) disturbed by arrivals from both streams. In subsequent slots it is disturbed only by X-stream arrivals and departures, and eventually reaches steady-state. Intuitively, we state that in transient stage the system states in consecutive slots are related with a geometrical change in their difference.

**The approximation**

Let $c_k(q)$ $(= P[C_k = q])$,where $q = (q_v, q_x, q_s, q_j)$ be the p.d.f. of system state at the beginning of slot k. Let $\delta_k(q)$ represent the difference in p.d.f. of system states in consecutive slots, and be defined as

$$\delta_k(q) = c_{k+1}(q) - c_k(q) \qquad \forall q, k$$

As our approximation, we state that for some finite value $T > 0$ , the system obeys the following relation

$$\delta_{k+1}(q) \;=\; \alpha\, \delta_k(q) \qquad \forall k \geq T \tag{23}$$

where $\alpha$ is a non-negative constant and satisfies $0 < \alpha < 1$. For a valid approximation we choose $T$ such that $0 \leq c_T(q) + \frac{\delta_T(q)}{(1-\alpha)} \leq 1$, $\forall q$. Using this approximation we can now represent the system state at the beginning of any slot $(> T)$ as

$$\begin{aligned}
c_{k+1}(q) &= c_k(q) \;+\; \delta_k(q) \\
c_{k+2}(q) &= c_{k+1}(q) \;+\; \delta_{k+1}(q) \\
&= c_k(q) \;+\; \{1+\alpha\}\delta_k(q) \\
c_{k+i}(q) &= c_k(q) \;+\; \{1+\alpha+\cdots+\alpha^{i-1}\}\delta_k(q) \\
&= c_k(q) + \frac{(1-\alpha^i)}{(1-\alpha}\delta_k(q), \qquad \forall q,\ \forall k \geq T,\ i \geq 0,\ 0 \leq j < N_S
\end{aligned} \tag{24}$$

For any m-state MMBP with one state as completely bursty (i.e. probability of arrival in this state is 1), the following is true of its p.d.f.

$$a_v(v', k+1) \;=\; \beta\, a_v(v', k) \qquad \forall k \geq m$$

where $\beta$ is a constant $(0 < \beta < 1)$. Using this property of an MMBP and the above stated approximation we can approximate the evolution step which is given as

$$c_0^{(n+1)}(q') = \sum_{k=1}^{\infty} \sum_{q,v'} a_v(v',k)\ c_k(n)(q) \tag{25}$$

where $q' = (v', x, s, j)$ and $q = (v, x, s, j)$. Let $k \geq T$ be the slots where we like to approximate instead of computing the exact system state. We can rewrite the evolution step as

$$c_0^{(n+1)}(q') = \sum_{k=1}^{T-1} \sum_{q,v'} a_v(v',k)\ c_k(n)(q) \sum_{k=T}^{\infty} \sum_{q,v'} a_v(v',k)\ c_k(n)(q) \tag{26}$$

By choosing a value of $T \geq m$, we can reduce the second summation in the above equation as

$$\begin{aligned}
\sum_{k=T}^{\infty} \sum_{q,v'} a_v(v',k)\ c_k(n)(q) &= \sum_{i=0}^{\infty} \beta^i a_v(v',T) \left( c_T(q) + \frac{(1-\alpha^i)}{(1-\alpha)} \delta_T(q) \right) \\
&= a_v(v',T) \left( c_T(q) \sum_{i=0}^{\infty} \beta^i + \frac{\delta_T(q)}{(1-\alpha)} \sum_{i=0}^{\infty} \beta^i (1-\alpha^i) \right) \\
&= \frac{a_v(v',T)}{(1-\beta)} \left( c_T(q) + \frac{\beta\ \delta_T(q)}{(1-\alpha\beta)} \right)
\end{aligned} \tag{27}$$

Substituting the above we can rewrite the approximated evolution step as follows

$$c_0^{(n+1)}(q') = \sum_{k=1}^{T-1} \sum_{q,v'} a_v(v',k)\ c_k(n)(q) + \frac{a_v(v',T)}{(1-\beta)} \left( c_T^{(n)}(q) + \frac{\beta\ \delta_T^{(n)}(q)}{(1-\alpha^{(n)}\beta)} \right) \tag{28}$$

For each evolution step we obtain $\delta_T^{(n)}(q)$ and $\alpha^{(n)}$ from $c_T^{(n)}(q)$, $c_{T+1}^{(n)}(q)$ and $c_{T+2}^{(n)}(q)$ (these are p.d.f.s of system states got using exact relations) as

$\delta_T^{(n)}(q) \ = \ c_{T+1}^{(n)}(q) \ - \ c_T^{(n)}(q)$ and

$\alpha_j^{(n)}(q) \ = \ \frac{c_{T+2}^{(n)}(q) \ - \ c_{T+1}^{(n)}(q)}{\delta_T^{(n)}(q)}.$

We consider $\alpha^{(n)}$ to be a weighted average which is calculated as

$$\alpha^{(n)} = \sum_{q} c_T^{(n)}(q)\ \alpha^{(n)}(q)$$

We could also obtain $\alpha^{(n)}$ as a simple average over all valid system states or as a weighted average over $\delta_T^{(n)}(q)$.

### 6.2.2 Case 2: Heterogeneous speed links ($N_S > 1$)

Here we consider the case where the output link is slower than the input link used by the V-stream. While considering homogeneous speed links, in our intuitive reasoning for the approximation we related the change in system state state in two consecutive slots. When we have heterogeneous speed links ($N_S > 1$), departures occur $N_S$ slots apart when the system is not empty and X-stream arrival can occur in any slot. The departure events occur more frequently and hence have more influence on the system state change than the X-stream arrivals which occur in batches. Intuitively, we state that in transient stage the system states every $N_S$ slots apart are related with a geometrical change in their difference.

**The approximation**

Let $c_k(q)$ ($= P[C_k = q]$),where $q = (q_v, q_x, q_s, q_j)$ be the p.d.f. of system state at the beginning of slot k. Let $\delta_k(q)$ represent the difference in p.d.f. of system states $N_S$ slots apart, and be defined as

$$\delta_k(q) = c_{k+N_S}(q) - c_k(q) \qquad \forall q, k$$

As our approximation, we state that for some finite value $T > 0$ , the system obeys the following relation

$$\delta_{k+N_S}(q) \;=\; \alpha\, \delta_k(q) \qquad \forall k \geq T \tag{29}$$

where $\alpha$ is a non-negative constant and satisfies $0 < \alpha < 1$. For a valid approximation we choose $T$ such that $0 \leq c_T(q) + \frac{\delta_T(q)}{(1-\alpha)} \leq 1$, $\forall q$. We consider the system states in the tail (i.e.$\geq T$) as $N_S$ geometric streams i.e., $\{c_{T+N_S i}(q)\}$, $\{c_{T+1+N_S i}(q)\}$, $\cdots$, $\{c_{T+N_S-1+N_S i}(q)\}$. Each stream ($j$) is treated separately with respect to the approximation and hence we have a constant $\alpha$ for each stream which is denoted as $\alpha_j$ where $0 \leq j < N_S$. Using this approximation we can now represent the system state at the beginning of any slot ($\geq T$) as

$$c_{k+j+N_S i}(q) \;=\; c_{k+j}(q) \;+\; \frac{(1-\alpha_j^i)}{(1-\alpha_j)}\,\delta_{k+j}(q), \quad \forall q,\ \forall k \geq T,\ i \geq 0,\ 0 \leq j < N_S \qquad (30)$$

For any m-state MMBP with one state as completely bursty (i.e. probability of arrival in this state is 1), the following is true of its p.d.f.

$$a_v(v', k+1) \;=\; \beta\, a_v(v', k) \qquad \forall k \geq m$$

where $\beta$ is a constant ($0 < \beta < 1$). Using this property of an MMBP and the above stated approximation we can reduce and rewrite the evolution step (equation (7)) as

$$\begin{aligned} c_0^{(n+1)}(q') \;=\; & \sum_{k=1}^{T-1} \sum_{q,v'} a_v(v',k)\; c_k^{(n)}(q) \\ & + \sum_{j=0}^{N_S-1} \frac{a_v(v', T+j)}{(1-\beta_S^N)} \left( c_{T+j}^{(n)}(q) \;+\; \frac{\beta_S^N\, \delta_{T+j}^{(n)}(q)}{(1-\alpha_j^{(n)}\beta_S^N)} \right) \end{aligned} \qquad (31)$$

For each evolution step we compute $\delta_{T+j}^{(n)}(q)$ and $\alpha_j^{(n)}$ from $c_{T+j}^{(n)}(q)$, $c_{T+j+N_S}^{(n)}(q)$ and $c_{T+j+2N_S}^{(n)}(q)$ (these are p.d.f.s of system states got using exact relations) as

$\delta_{T+j}^{(n)}(q) \;=\; c_{T+j+N_S}^{(n)}(q) \;-\; c_{T+j}^{(n)}(q)$ and $\alpha_j^{(n)}(q) \;=\; \frac{c_{T+j+2N_S}^{(n)}(q) \;-\; c_{T+j+N_S}^{(n)}(q)}{\delta_{T+j}^{(n)}(q)}$, where $0 \leq j < N_S$.

We consider $\alpha_j$ to be a weighted average which is calculated as $\alpha_j^{(n)} \;=\; \sum_q c_{T+j}^{(n)}(q)\; \alpha_j^{(n)}(q)$.

# 7 Numerical Results

In this section we study the effect of traffic characteristics and buffer sizing on the performance of a session. We compare some of these results with simulation results to validate the accuracy of our analysis and the approximation technique. We consider a scenario where a session's offered load is 20% of an output link capacity while the cross traffic requires 40%.

We present results for two cases ($N_A = 1$ in both cases):

(1) both input and output links are of same speed ($N_S = 1$), and

(2) packets of a session arrive four times faster than they can be transmitted ($N_S = 4$), i.e., input and output links may be using SONET OC-12c and OC-3c protocols respectively.

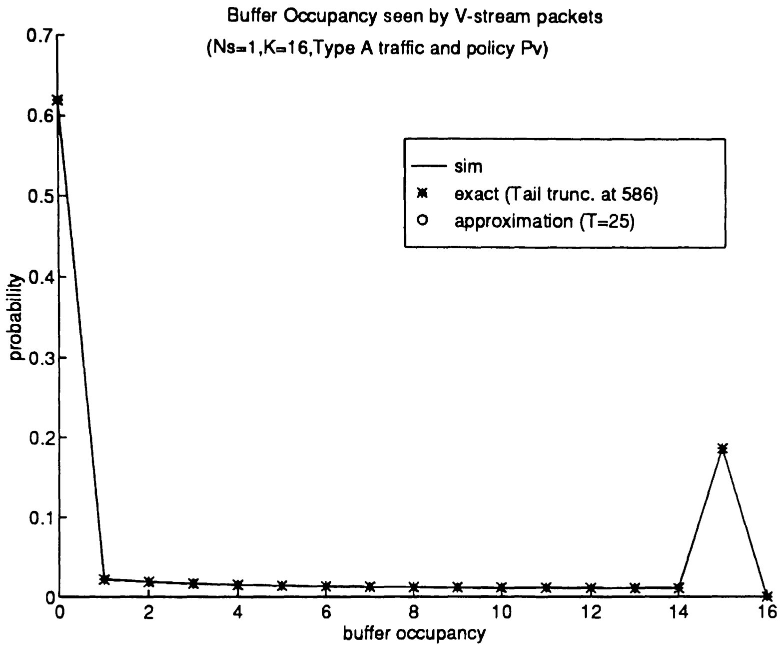
We consider different traffic types based on burstiness and correlation of the two streams; they are tabulated below in table 2.

| Traffic type | *V-stream (offered load ($\lambda N_S$)=0.2)* | | | | *X-stream (offered load ($\lambda N_S \bar{B}$)=0.4)* | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | bursty | $C^2$ | correlated | $\psi_1$ | bursty | $C^2$ | correlated | $\psi_1$ | Batch | $\bar{B}$ |
| A | √ | 16 | √ | 0.1 | √ (high) | 16 | √ | 0.1 | Geo | 3 |
| B | √ | 16 | √ | 0.1 | √ (low) | 1 | X | 0 | Geo | 3 |
| C | √ | 16 | √ | 0.1 | X | 1 | X | 0 | const. | 1 |
| D | √ | 16 | X | 0 | X | 1 | X | 0 | const. | 1 |
| E | X | 1 | X | 0 | X | 1 | X | 0 | const. | 1 |

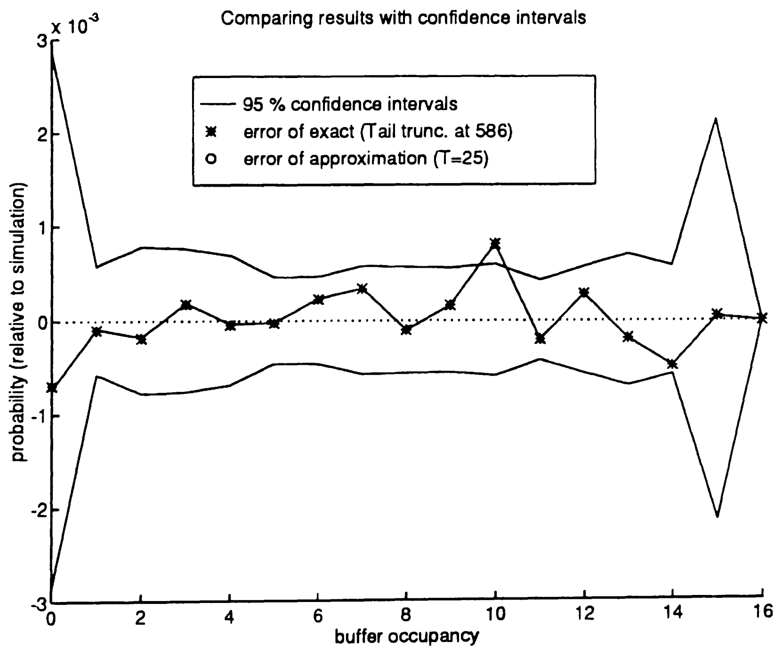
Table 2: Traffic parameters for V-stream and X-stream

## 7.1 Validating the accuracy of results

To validate our results obtained using exact analysis (by tail truncation) and results obtained using geometric tail approximation, we compare them with simulation results. We consider three sets of results using the approximation technique. Figure 4(a) shows the buffer occupancy seen by V-stream where traffic is of type A, policy is $\mathcal{P}_V$ and buffer size $K = 16$. Notice that there is no difference between the simulation and analytical results; for a better view of the comparison, we plot the absolute error (= *analytical(x) - simulation(x)* ) along with the $95^{th}$ % confidence regions of simulation results. For this example, tail truncation of arrival process was at $M_A = 560$ slots at an accuracy of $10^{-6}$. We see that, approximation with $T = 25$ slots yields near exact results; a reduction of computational time by a factor of 23. Gain in computational cost decreases with increase of $N_S$ and buffer size. In figure 5, we plot the buffer occupancy and the observed error when the policy is $\mathcal{P}_X$; here we consider

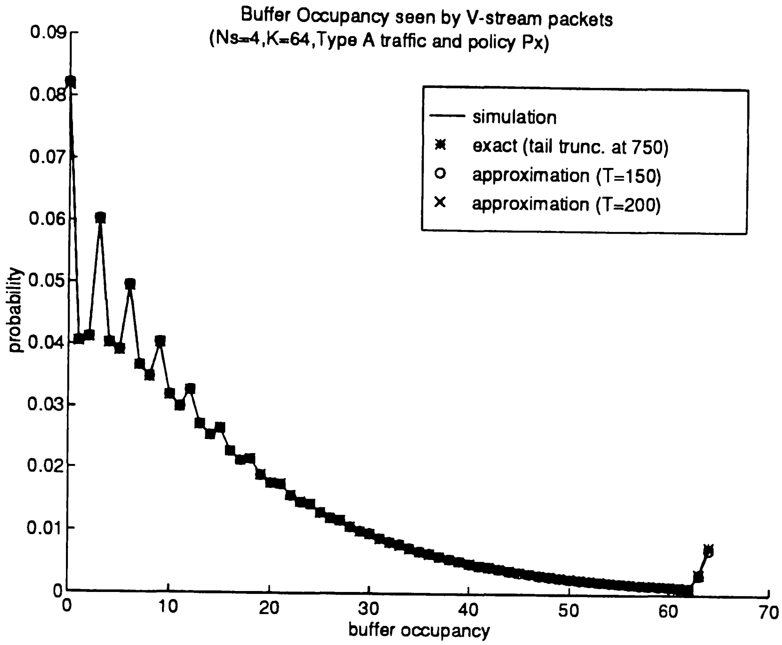



(a) Buffer occupancy distribution

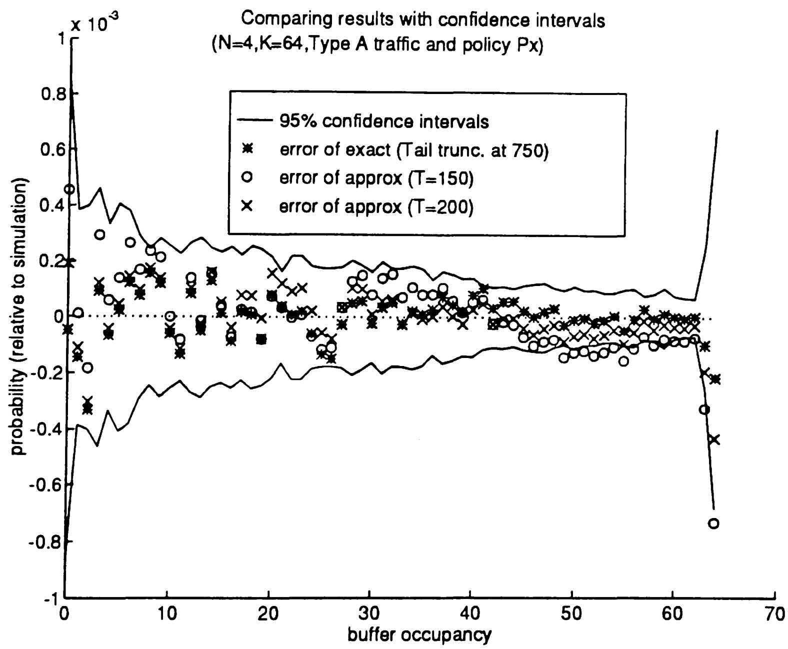



(b) Error graph

Figure 4: Comparison for accuracy of approximation: Buffer occupancy seen by V-stream : $N_S = 1$, $K = 16$ system with type A traffic and policy $\mathcal{P}_V$




(a) Buffer occupancy distribution




(b) Error graph

Figure 5: Comparison for accuracy of approximation: Buffer occupancy seen by V-stream : $N_S = 4$, $K = 64$ system with Type A traffic and policy $\mathcal{P}_X$

an $N_S = 4$, $K = 64$ system with type B traffic. Here, since the arrival process is bursty and has low mean arrival rate ($\lambda = 0.05, C^2 = 16, \psi_1 = 0.1$) we would need $M_A = 2061$ for an accuracy $\delta = 10^{-6}$. Instead, we truncate the tail at 750 slots (at an accuracy of 1.53e-3) to see the influence on results. We notice that these results and approximation ($T = 200$), are within the confidence region. These two figures show that using the approximation technique we can obtain our results efficiently while maintaining their accuracy.

## 7.2 Effect of burstiness and correlation

First, we illustrate an important phenomenon which occurs in $N_S > 1$ systems when the traffic is bursty. In figure 5, notice that the probability values when the buffer occupancy is a multiple of $(N_S - 1)$ is higher than its neighboring values. It can be explained using an example scenario: Consider a busy period of the system where the first $n$ ($n > N_S$)arrivals are from V-stream and all of which occur in consecutive slots (see figure 6). The first packet

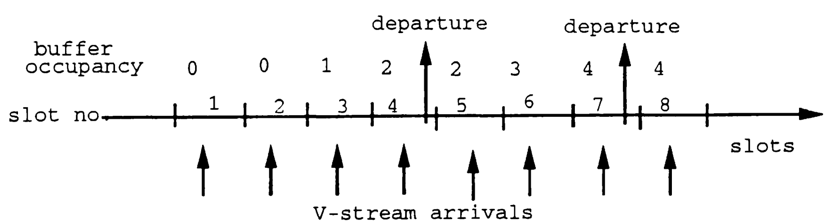



Figure 6: A sample busy period with V-stream arrivals occurring in the first $n$ consecutive slots and no cross traffic

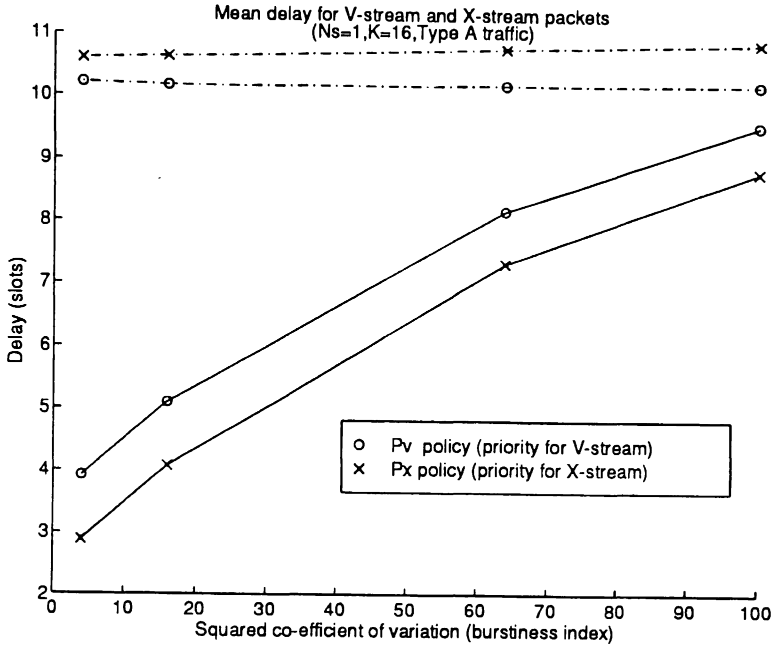
begins service in slot 2 and departs at the end of slot 5 (since $N_S = 4$). The arrivals which occur in slot 5 and 6 respectively see the buffer occupancy to be 4. Also, for each departure $i$ which occurs at the end of slot $iN_S + 1$ in this interval of $n$ arrivals, the occupancy seen by an arrival in slots $iN_S + 1$ and $iN_S + 2$ is $i(N_S - 1)$ packets; hence an higher probability values for $i(N_S - 1)$. Such scenarios are very likely to occur since both streams are bursty. This behavior is also reflected in the departure process; it introduces periodic (or negative) correlation between departures which makes the characterization of the departure process difficult.

In figures 7(a) and 7(b) we plot the mean delay and loss probability of V-stream and X-stream for change in V-stream burstiness values for a system where $N_S = 1$, $K = 16$, with type A traffic. We notice that the system is in a high-loss state with up to 50% of X-stream packets being lost since X-stream is very bursty and buffer size is small. Under these conditions the change in X-stream performance is negligible; but V-stream delay increases considerably especially when it is initially not bursty. By observing these two streams we state that sources which are not-bursty are more sensitive to burstiness than those which are already bursty. The converse seems true for the effect of autocorrelation (see figure 8), i.e., correlated streams are more sensitive than non-correlated streams, for change in autocorrelation.

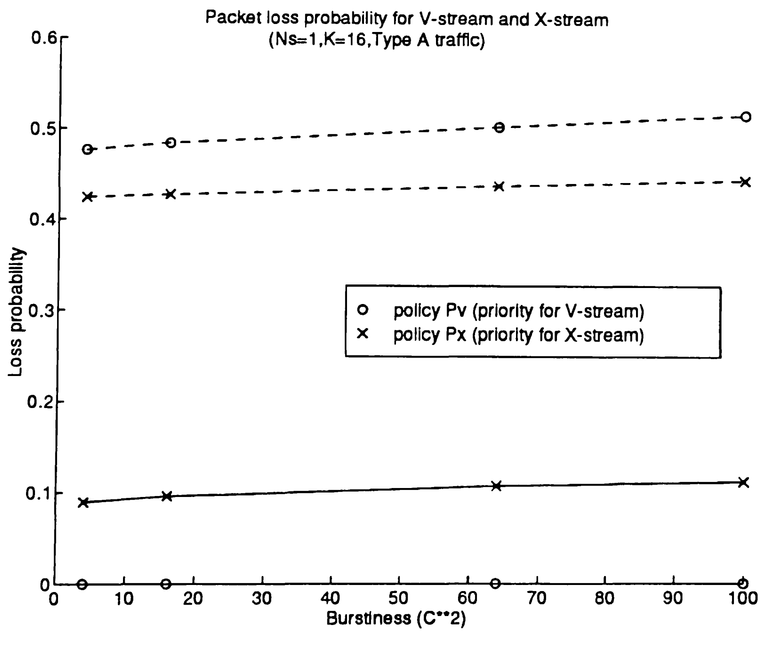
## 7.3 Influence of slot policy ($\mathcal{P}$) on performance

In figures 4(a) and 7(b) we notice that V-stream packets are not lost when the policy is $\mathcal{P}_V$, since in each slot there is always space for one packet in the buffer which is made available at the beginning of a slot when a packet starts its service. If policy is $\mathcal{P}_X$, we notice that if the arrivals are bursty and correlated then packet loss probability is high; resulting in a high variability between these two policies. We also notice such behavior with delay in figure 7(a) where the mean delay changes by 25% in the worst case. When the system is more stable (less loss and has larger buffer sizes) or when $N_S > 0$, this behavior diminishes and becomes almost negligible. This high variability in performance measures due to the slot policy indicates that the port of arrival at a switching element is an important factor which influences the performance of a session when we have homogeneous speed links. A possible solution to avoid such behavior is to either add intelligence (or randomize) to change the polling sequence with time for switching, or randomly distribute the packets at the input ports.

Notice in figures 7 and 8 that V-stream has a larger mean delay when the policy is $\mathcal{P}_V$, which contradicts a belief that a better delay performance should result from slot priority.

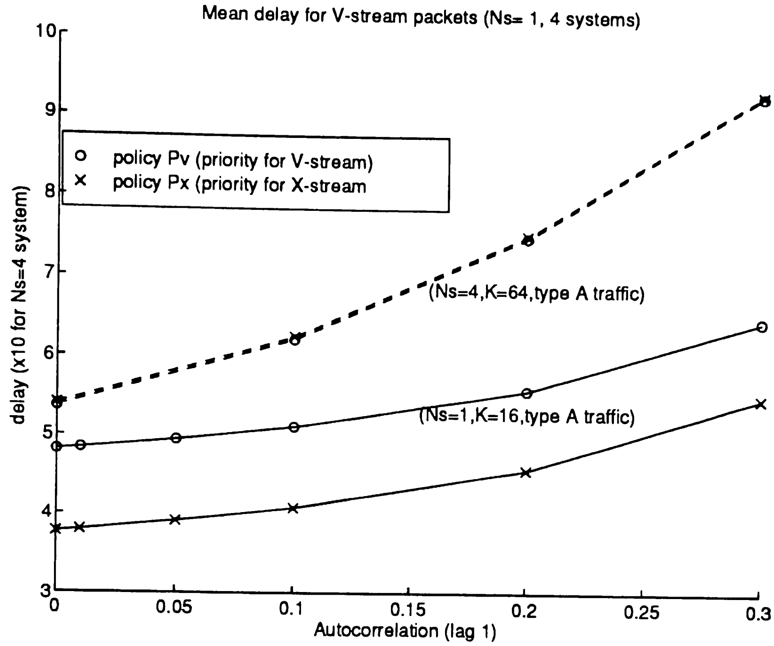



(a) Mean delay

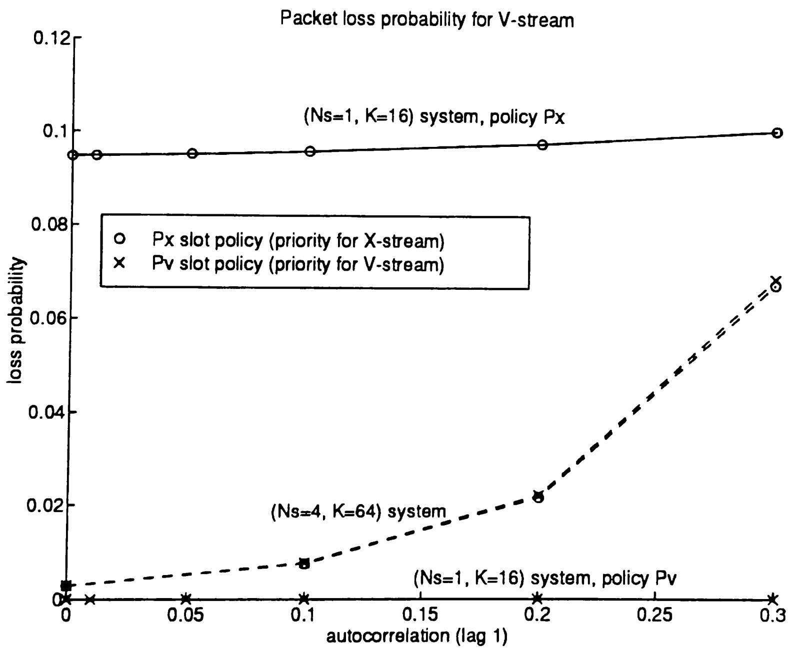



(b) Packet loss probability

Figure 7: Effect of change in burstiness of V-stream on mean delay and loss probability of both the streams: $N_S = 1, K = 16$ system with Type A traffic




(a) Mean delay




(b) Packet loss probability

Figure 8: Effect of change in autocorrelation of V-stream on its mean delay and loss probability: $(N_S = 1, K = 16)$ and $(N_S = 4, K = 64)$ systems with Type A traffic

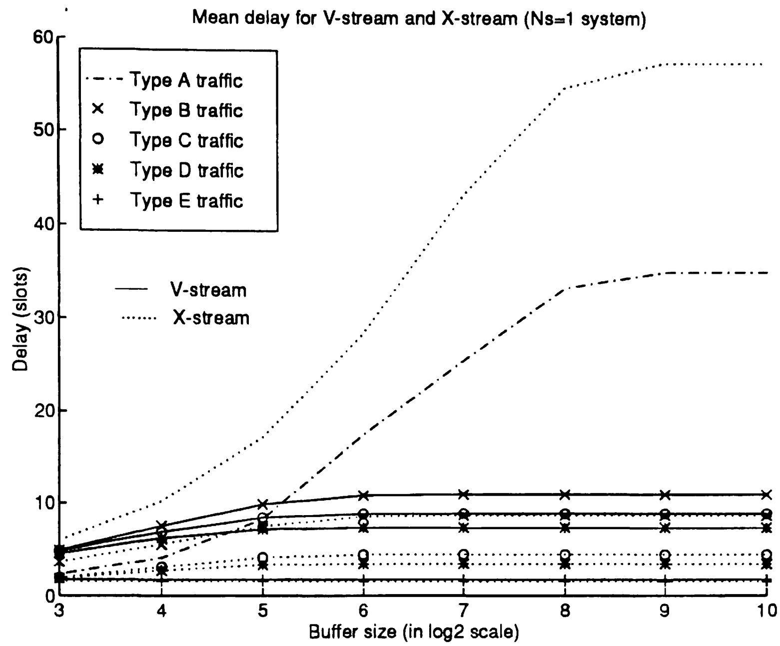
This can be explained as follows: delay is calculated for packets which enter the system. All the packets which are lost if policy is $\mathcal{P}_X$, find place in buffer when the policy is $\mathcal{P}_V$, even during temporary congestion states (due to bursty arrivals). These packets incur larger delays than the mean delay when policy is $\mathcal{P}_X$; hence an increased mean delay when policy is $\mathcal{P}_V$.

### 7.4 Influence of buffer size

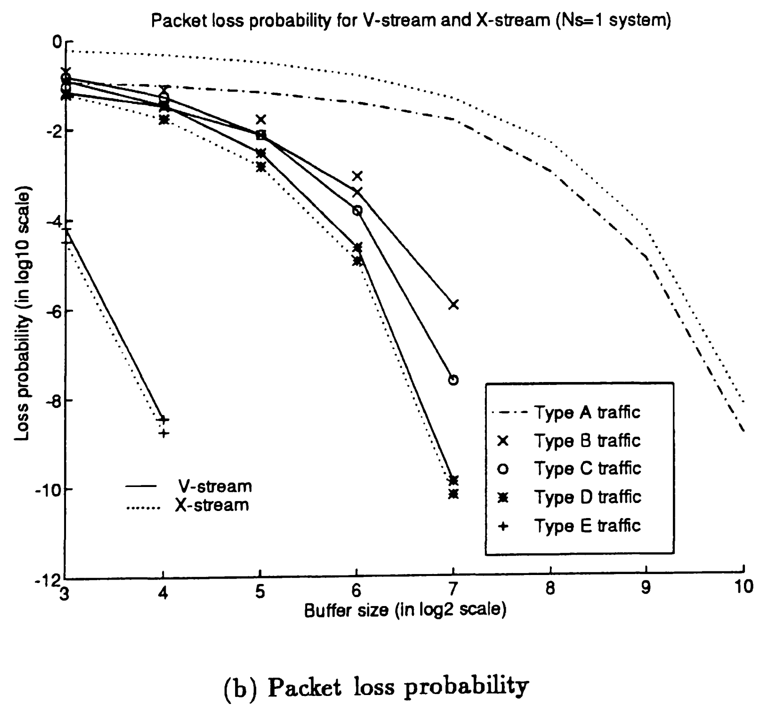
An important aspect of designing a switch is buffer dimensioning since loss guarantees may be of the order $< 10^{-10}$. Obtaining accurate results of such small magnitudes through simulation techniques is clearly a difficult (time consuming) task; which makes our analysis more useful in studying systems with low loss probabilities for exact results. In figures 9 and 10, we plot the mean delays and loss probability for both $N_S = 1$ and $N_S = 4$ systems with buffer size ranging from $2^3$ to $2^{10}$ packets. We consider all traffic types listed in table 2. We notice that (for the same type of traffic) $N_S = 4$ system has larger buffer requirements while both show a similar trend. When the V-stream and X-stream are bursty and correlated (type A traffic) we notice that even for a small gain in loss probability we need a large increase in buffer size; on increasing buffer size we note that the delay increases (until the loss probabilities are very small). There seems to be a trade-off between the two, which suggests that use of priority policies for service (rather than FIFO service policy) will help in optimizing the performance of streams. It will be efficient especially when a QoS requested by a stream is one of the two (delay or packet loss).

## 8 Conclusions

In this paper we have presented exact analyses for per-session analysis of an ATM switching node with two important characteristics: (1) the traffic streams are bursty and the inter-arrival times are correlated, and (2) the links across the switch may be of different speeds. The switching node was modeled as a discrete-time finite capacity queue with two super-

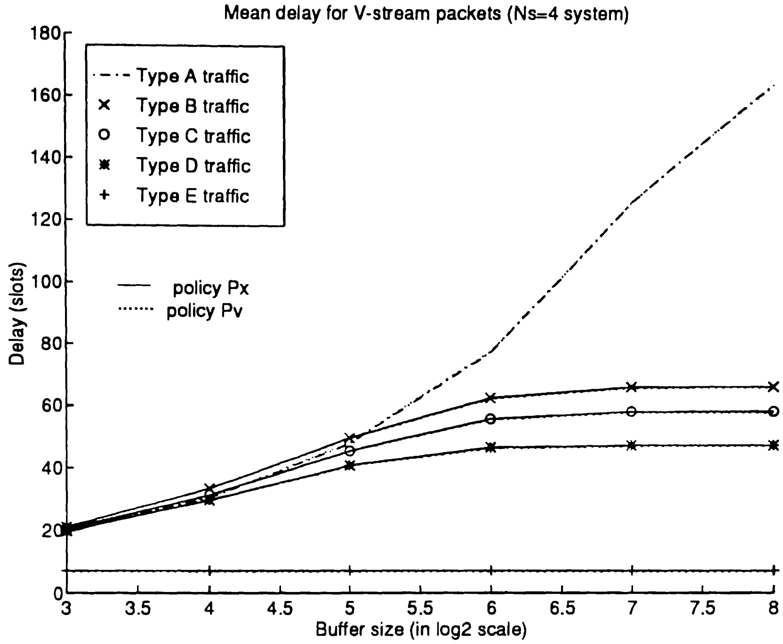



(a) Mean delay

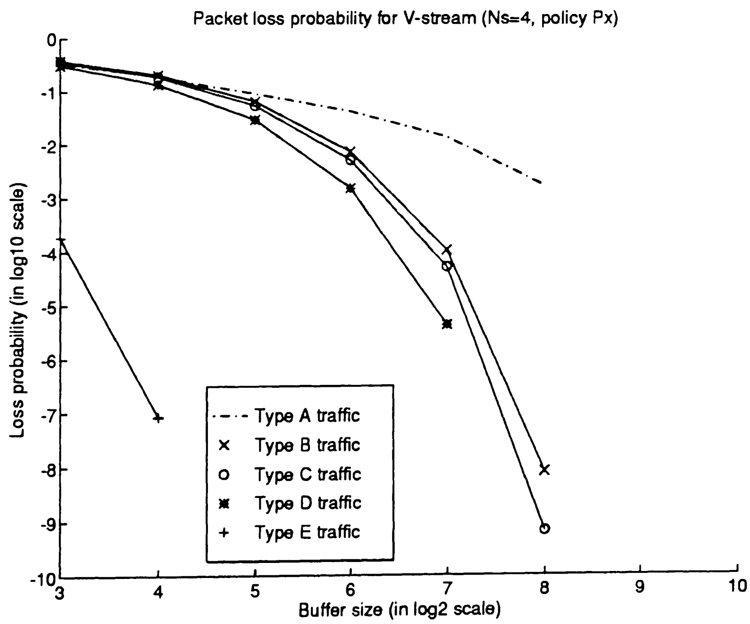



(b) Packet loss probability

Figure 9: Influence of change in buffer size on mean delay and loss probability of V-stream and X-stream: $N_S = 1$ system with all types of traffic




(a) Mean delay




(b) Packet loss probability

Figure 10: Influence of change in buffer size on mean delay and loss probability of V-stream: $N_S = 4$ system with all types of traffic

posing non-renewal arrival processes. We obtain the quality of service (QoS) measures of a single session of interest and the cross traffic. Even though we used MMBP as our traffic model for a session, the method can be easily extended to other traffic models. We also presented an approximation technique to reduce the computational time; we have shown it to be very accurate and efficient by comparing with simulation results. For an example switching scenario we also illustrated the effects of traffic characteristics, port of arrival of a session and buffer size on the steady-state performance measures of the traffic streams. This method (with some simple modifications) could also be used to study the transient performance of sessions. The analysis presented in this paper is the first step towards studying the performance of a session across an ATM network. Future work includes characterization of departure process (at a switching node) and end-to-end analysis of a session.

# REFERENCES


[1] E. Cinlar. *Introduction to stochastic processes.* Prentice-Hall Inc., Englewood Cliffs, NJ, 1975.

[2] W. Fischer and K. Meier-Hellstern. The mmpp cookbook. *Performance Evaluation*, 18(2):149–171, Sept 1993.

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

[4] C. Herrmann. Correlation effect on per-stream qos parameters of atm traffic superpositions relevant to connection admission control. In *Proceedings, IEEE International Conference on Communications*, pages 1027–1031, May 1993.

[5] C. Johnston and H. Chao. The atm layer chip: An asic for b-isdn applications. *IEEE Journal on Selected Areas in Communications*, 9(5):741–750, June 1991.

[6] L. Kleinrock. *Queueing Systems*, volume 1. Wiley-Interscience, John Wiley & Sons, NY, 1975.

[7] D. Kong. 2.488 gb/s sonet multiplexer/demultiplexer with frame detection capability. *IEEE Journal on Selected Areas in Communications*, 9(5):726–731, June 1991.

[8] M. Murata, Y. Oie, T. Suda, and H. Miyahara. Analysis of discrete-time single server queue with bursty inputs for traffic control in atm networks. *IEEE J. Select. Areas Commun.*, 8(3):447–458, Apr 1990.

[9] Y. Ohba, M. Murata, and H. Miyahara. Analysis of interdeparture process for bursty traffic in atm networks. *IEEE Journal on Selected Areas in Communications*, 9(3):468–476, Apr 1991.

[10] C. Omidyar and A. Aldridge. Introduction to sdh/sonet. *IEEE Communications Magazine*, 31(9):30–33, June 1993.

[11] R. Onvural and I. Nikolaidis. Routing in atm networks. In H. Perros, editor, *Proceedings, TriComm '92: High-Speed Communications Networks*, pages 139–150, Feb 1992.

[12] D. Park. *MMBP characterization of the departure process of a discrete-time queue arising in ATM networks.* PhD thesis, Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, 1993. PhD Thesis.

[13] T. Robe and K. Walsh. A sonet sts-3c user network interface integrated circuit. *IEEE Journal on Selected Areas in Communications*, 9(5):732–740, June 1991.

[14] J. Roberts. *COST 224 - performance evaluation and design of multiservice networks.* Commission of the European Communities, Luxemburg, France, 1991.

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

# A Two state Markov Modulated Bernoulli Process (MMBP)

A 2-state MMBP is characterized by the state-transition probability matrix $P_t$ of the Markov chain and the arrival rate descriptor $\Lambda$ defined as follows:

$$\mathbf{P_t} = \begin{bmatrix} p & 1-p \\ 1-q & q \end{bmatrix} \quad \text{and} \quad \Lambda = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

The MMBP process stays in a particular state (state 1) for a period which is geometrically distributed during which arrivals occur in a Bernoulli fashion with a specific probability $\alpha$. This period is followed by another geometrically distributed period (state 2) during which arrivals occur in a Bernoulli fashion with a different probability $\beta$. These periods continuously alternate representing the state transition between the two states. Given that the MMBP is in state 1(or state 2) at slot i, it will remain in the same state in the next slot $i+1$ with probability (w.p.) $p$ (or $q$), or changes its state w.p. $1-p$(or $1-q$).

Let $t$ be the interarrival time between two successive arrivals, and $t_{ij}$ be the time interval starting from a particular slot when the arrival process is in state $i$ and ending at a slot when the next arrival occurs and the arrival process is in state $j$. The time interval $t_{ij}, 1 \leq i, j \leq 2$, can be expressed as the following:

$$t_{11} = \begin{cases} 1 & w.p. \quad \alpha p \\ 1 + t_{11} & w.p. \quad (1-\alpha)p \\ 1 + t_{21} & w.p. \quad (1-\beta)(1-p), \end{cases}$$

$$t_{21} = \begin{cases} 1 & w.p. \quad \alpha(1-q) \\ 1 + t_{11} & w.p. \quad (1-\alpha)(1-q) \\ 1 + t_{21} & w.p. \quad (1-\beta)q, \end{cases}$$

$$t_{12} = \begin{cases} 1 & w.p. \quad \beta(1-p) \\ 1 + t_{12} & w.p. \quad (1-\alpha)p \\ 1 + t_{22} & w.p. \quad (1-\beta)(1-p), \end{cases}$$

$$t_{22} = \begin{cases} 1 & w.p. \quad \beta q \\ 1 + t_{12} & w.p. \quad (1-\alpha)(1-q) \\ 1 + t_{22} & w.p. \quad (1-\beta)(1-q). \end{cases}$$

Let $S_n$ be the state of the MMBP when the $n^{th}$ arrival occurs, and let $T_{n,j}$ be the interarrival time between the $(n-1)^{th}$ and $n^{th}$ arrivals while the $n^{th}$ arrival occurs in state $j$. If we define

$$A_{ij}(z) \equiv E[z^{T_{n,j}} | S_{n-1} = i],$$

then from the definition of $t_{ij}$ and $T_{n,j}$ we have

$$A_{ij}(z) = E[z^{t_{ij}}], \qquad \text{where } 1 \le i, j \le 2.$$

Therefore,

$$A_{11}(z) = \alpha pz + (1-\alpha)pzA_{11}(z) + (1-\beta)(1-p)zA_{21}(z), \tag{32}$$

$$A_{21}(z) = \alpha(1-q)z + (1-\alpha)(1-q)zA_{11}(z) + (1-\beta)qzA_{21}(z), \tag{33}$$

$$A_{12}(z) = \beta(1-p)z + (1-\alpha)pzA_{12}(z) + (1-\beta)(1-p)zA_{22}(z), \tag{34}$$

$$A_{22}(z) = \beta qz + (1-\alpha)(1-q)zA_{12}(z) + (1-\beta)qzA_{22}(z). \tag{35}$$

By sloving equations (32) - (35), we have

$$A_{11}(z) = \frac{\alpha pz + \alpha(1-\beta)(1-p-q)z^2}{1 - ((1-\alpha)p + (1-\beta)q)z - (1-\alpha)(1-\beta)(1-p-q)z^2}, \tag{36}$$

$$A_{12}(z) = \frac{\beta(1-p)z}{1 - ((1-\alpha)p + (1-\beta)q)z - (1-\alpha)(1-\beta)(1-p-q)z^2}, \tag{37}$$

$$A_{21}(z) = \frac{\alpha(1-q)z}{1 - ((1-\alpha)p + (1-\beta)q)z - (1-\alpha)(1-\beta)(1-p-q)z^2}, \tag{38}$$

$$A_{22}(z) = \frac{\beta qz + (1-\alpha)\beta(1-p-q)z^2}{1 - ((1-\alpha)p + (1-\beta)q)z - (1-\alpha)(1-\beta)(1-p-q)z^2}. \tag{39}$$

Here $A_{ij}(z)$ represents the generating function of the conditional p.d.f. of interarrival time of VC-stream packets, $a_i(j,k)$. The generating functions of the p.d.f. of the interarrival time ($A(z)$) is given as:

$$Pr[t=k] = \sum_{j=1}^{2}\sum_{i=1}^{2} Pr[t_{ij}=k]\tilde{\pi}(i) \tag{40}$$

where $\tilde{\pi}(i)$ is the steady state probability of having an arrival in state $i$, and is given by $\tilde{\pi}=[\frac{\alpha(1-q)}{\alpha(1-q)+\beta(1-p)}, \frac{\beta(1-p)}{\alpha(1-q)+\beta(1-p)}]$. By applying z-transform to the above equation, $A(z) \equiv E\{z^t\}$ is given as

$$A(z) = \frac{\alpha(1-q)(A_{11}(z)+A_{12}(z))+\beta(1-p)(A_{21}(z)+A_{22}(z))}{\alpha(1-q)+\beta(1-p)}. \tag{41}$$

We obtain the interarrival time distribution $a(i)$ (=Pr[Interarrival time = i slots]) by inverting the generating functions $A(z)$.

$$A(z) = \frac{\sum_{i=0}^{2} b_i z^i}{\sum_{i=0}^{2} c_i z^i} = \sum_{i=0}^{\infty} a(i) z^i.$$

Multiplying both sides by the denominator and then equating the coefficients of $z^i$ on both sides we have the following set of linear difference equations:

$$\sum_{j=0}^{min\{2,i\}} c_i a(i-j) = \begin{cases} b_i & i=0,1,2 \\ 0 & i>2 \end{cases}$$

We can now obtain the distribution $a(i), i = 0, 1, ...$, using the recurrence relation:

$$a(i) = \frac{1}{c_0}\left[\ [b_i - \sum_{j=1}^{min\{2,i\}} c_i a(i-j)\ \right], \qquad i=0,1,...,$$

where $b_i = 0$ for $i > 2$.

Similarly we obtain the conditional interarrival time p.d.f. $a_v(v', k)$ by inverting the generating functions $A_{vv'}(z)$, $1 \le v, v' \le 2$ individually.

For an MMBP, the mean arrival rate $\lambda$ ($= 1/E\{t\}$) and the squared coefficient of variation of the time between successive arrivals $C^2$ are:

$$\lambda = \frac{\alpha(1-q)+\beta(1-p)}{2-p-q}, \tag{42}$$

$$
\begin{aligned}
C^2 &= \frac{Var(t)}{E\{t\}^2} \\
&= \frac{2[\alpha(1-q)+\beta(1-p)]}{\alpha(1-q)+\beta(1-p)+\alpha\beta(p+q-1)} - \frac{\alpha(1-q)+\beta(1-p)}{2-p-q} \\
&\quad + \frac{2[\alpha(1-p)+\beta(1-q)][\alpha(1-q)+\beta(1-p)](p+q-1)}{(2-p-q)^2[\alpha(1-q)+\beta(1-p)+\alpha\beta(p+q-1)]} - 1.
\end{aligned} \tag{43}
$$

The autocorrelation coefficient of the interarrival time of MMBP with lag 1 is given by

$$
\begin{aligned}
\psi_1 &= \frac{Cov(T_{n-1}\, T_n)}{Var(T_n)} = \frac{E[T_{n-1}\, T_n] - E(T_{n-1})E(T_n)}{Var(T_n)} \\
&= \frac{\alpha\beta(\alpha-\beta)^2(1-p)(1-q)(p+q-1)^2}{C^2(2-p-q)^2[\alpha(1-q)+\beta(1-p)+\alpha\beta(p+q-1)]^2}.
\end{aligned} \tag{44}
$$

## A.1 Obtaining 2-MMBP from traffic descriptors $\lambda$, $C^2$ and $\psi_1$

For given values of $\lambda$, $C^2$ and $\psi_1$, we describe a simple way to invert the equations (??), (43) and (44) to obtain the transition probability matrix ($P_t$) and arrival rate descriptor $\Lambda$. We fix the value of $\alpha$, $0 < \alpha \leq 1$, and obtain the values for $\beta$, $p$ and $q$ using the above three equations. The key to our solution is to consider the factor '$\bar{p}+\bar{q}$' as the common working variable to reduce the above equations. Using equation (42) we can reduce equation (43) to

$$
\bar{p}+\bar{q} = \frac{\beta+G}{\beta+G+D} \tag{45}
$$

$$
\textit{where} \qquad G = \frac{\alpha-\lambda}{1-\alpha A}, \quad D = \frac{A\lambda-1}{1-\alpha A}, \quad \textit{and} \quad A = \frac{C^2+\lambda+1}{2\lambda}.
$$

Using equations (42) and (45) we can now reduce equation (44) to a quadratic equation of variable $\beta$ i.e., $a_2\beta^2 + a_1\beta + a_0 = 0$, where

$$
\begin{aligned}
a_0 &= (\lambda G)^2 \\
a_1 &= 2\lambda G(\lambda + D\alpha) - E\lambda \\
a_2 &= (\lambda + D\alpha)^2 + E \\
\textit{and} \;\; E &= \frac{\alpha(\alpha-\lambda)D^2}{\psi_1 C^2}.
\end{aligned} \tag{46}
$$

The roots of the above quadratic equation are

$$\beta = \frac{-a_1 \pm \sqrt{a_1^2 - 4a_0a_2}}{2a_2} \tag{47}$$

The obtained solution is valid if $\beta$ satisfies $0 < \beta \leq 1$. The values of $p$ and $q$ are given as

$$p = 1 - \frac{(\alpha - \lambda)}{(\alpha - \beta)} \frac{(\beta + G)}{(\beta + G + D)} \tag{48}$$

$$q = 2 - p - \frac{\beta + G}{\beta + G + D}. \tag{49}$$