Sizing Exit Buffers in ATM Networks: An Intriguing Coexistence of Instability and Tiny Cell Loss Rates

Hanoch Levy, *Member, IEEE,* Tzippi Mendelson, Moshe Sidi, *Senior Member, IEEE,* and Joseph Keren-Zvi, *Associate Member, IEEE*

***Abstract*—This paper deals with the sizing of end buffers in ATM networks for sessions subject to constant bit rate (CBR) traffic. Our objective is to predict the cell-loss rate at the end buffer as a function of the system parameters. We introduce the D+G/D/1 queue as a generic model to represent exit buffers in telecommunications networks under constant rate traffic, and use it to model the end buffer. This is a queue whose arrival rate is equal to its service rate and whose arrivals are generated at regular intervals and materialize after a generally distributed random amount of time. We reveal that under the infinite buffer assumption, the system possesses rather intriguing properties: on the one hand, the system is *instable* in the sense that the buffer content is monotonically nondecreasing as a function of time. On the other hand, the likelihood that the buffer contents will exceed certain level B by time t diminishes with B. Improper simulation of such systems may therefore lead to false results. We turn to analyze this system under finite buffer assumption and derive bounds on the cell-loss rates. The bounds are expressed in terms of simple formulae of the system parameters. We carry out the analysis for two major types of networks: 1) datagram networks, where the packets (cells) traverse the network via independent paths and 2) virtual circuit networks, where all cells of a connection traverse the same path. Numerical examination of ATM-like examples show that the bounds are very good for practical prediction of cell loss and the selection of buffer size.**

***Index Terms*—ATM, buffer sizing, CBR, D+G/D/1 queue, end-to-end loss rate.**

I. Introduction

THE SUBJECT of this paper is the analysis of end buffers in ATM networks when they are subject to constant bit rate (CBR) traffic. Our major objective is the derivation of the cell loss rate at the end buffers, as to yield proper design of these buffers.

In ATM networks with CBR traffic, as in a variety of other data and telecommunication networks, a session typically consists of a data source node and a destination node whose processing/transmission speeds are *identical* to each other and which are connected by a network whose speed and delay can be arbitrary.

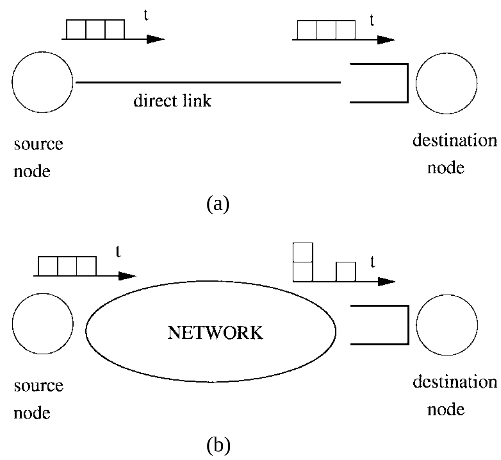


Fig. 1. Connection of source and destination nodes. Connection via (a) a direct link and (b) an arbitrary network.

In a communications system where a data source node is connected *directly* to the destination node [see Fig. 1(a)], no buffers are required at the destination node (or one buffer suffices) since the speeds of the destination and the source are identical.

In contrast, in systems where the connection between the source node and the destination node are done via an arbitrary network [see Fig. 1(b)], the delays incurred by the data units are arbitrary and, thus, buffers are required at the destination node to avoid data loss. A major design issue is that of selecting the size of the buffer at the destination node so as to provide that data loss will not exceed desired values.

The objective of this paper is the derivation of data-loss measures at the exit buffer as a function of the network parameters. In particular, such derivation is to be conducted as a function of the delay incurred by the cells while passing through the network and which is assumed to be derived by other means. Our interest is in the size of the buffer required at the end buffer to provide that the probability of cell loss will not exceed certain values.

We assume that the source generates a continuous stream of cells (packets) whose lengths are all identical to each other. Thus, the source generates cells at a deterministic rate of one cell per unit of time and the processing time of each cell is deterministic (of one unit). The destination node has the same

Manuscript received March 16, 1998; approved by IEEE/ACM Transactions on Networking Editor T. V. Lakshman. This work was supported by the Consortium for Broadband Communication under the Chief Scientist of the Israeli Ministry of Commerce and Industry. A earlier version of this paper appeared in the *Proc. INFOCOM'98,* San Francisco, CA, pp. 1309–1316.

H. Levy is with the Department of Computer Science, Tel-Aviv University, Tel-Aviv, Israel, and ECI Telecom, Petach-Tikva, Israel (hanoch@math.tau.ac.il).

T. Mendelson was with ECI Telecom, Petach-Tikva, Israel. She is now with the Racah Institute of Physics, The Hebrew University, Jerusalem, Israel (e-mail: tsippora@shum.huji.ac.il or perachsheli@hotmail.com).

M. Sidi is with the Department of Electrical Engineering, Technion–Israel Institute of Technology, Haifa 32000, Israel.

J. Keren-Zvi is with ECI Telecom, Petach-Tikva, Israel (e-mail: joseph-jkzcom@email.msn.com).

Publisher Item Identifier S 1063-6692(99)09709-5.

processing power as the source node; thus the processing time of a cell at the destination node is one unit of time.

To analyze the behavior of the exit buffer, we introduce a new concept of arrival process, *deterministically-generated generally-postponed* arrivals and a new queueing model, the D+G/D/1 queue. In the deterministically generated generally postponed arrival process, arrivals are generated at deterministic (regular) intervals, but their arrival time to the queue is postponed by a generally distributed random variable, called the *postpone distribution*. Thus, if C_t is the cell generated at epoch t, then C_t enters the queue at $t+\mathcal{P}_t$, where $\mathcal{P}_t$ is a generally distributed random variable. The D+G/D/1 queue is a queue where the arrival process is deterministically distributed generally postponed, the service time is deterministic and there is a single server. A special case of the D+G/D/1 queue is the D/D/1 (the postpone value of each arrival is deterministically 0) which is trivial to analyze. The D+G/D/1 queue seems to be inherently different from either the D/G/1 queue or the G/D/1 queue; see remarks on this issue in Section I-A below.

Unfortunately, the structure of the D+G/D/1 queue seems to be too complicated to yield *exact* analysis of the buffer occupancy and loss rate. In particular, the use of infinite buffer analysis (which is a common technique) cannot deliver such results. Rather, our analysis (Section III) of the infinite buffer model reveals several intriguing properties: on the one hand, the system is *instable* in the sense that the buffer content is monotonically nondecreasing as a function of time. On the other hand, the likelihood that the buffer contents will exceed certain level B by time t, diminishes with B.

These properties suggest that the cell-loss analysis must be conducted *directly* on a finite buffer system, which we carry out in Section IV. We bound the cell-loss rate by both the postpone distribution and the postponed cell population distribution.

The derivation of practical bounds for cell loss is then carried out for two major network models. The first network model is a datagram service network, in which different cells of a session traverse the network independently of each other. This models networks similar to the Internet. In such networks, the postpone values of different cells (packets) are independent of each other (namely, $\mathcal{P}_t$ and $\mathcal{P}_s$ are independent for $s \neq t$). In Section V, we consider this model and derive the proper loss rates.

The second network model is a virtual circuit-like network where all cells of the same session follow exactly the same path. This models the behavior of ATM well. In such networks, the behavior of different cells in the session is not independent of that of others. In particular, they obey a simple constraint by which young cells cannot overtake older cells. The analysis of this model is provided in Section VI. A practical interpretation of these results is provided in Section VII.

Finally, in Section VIII, we conduct a numerical examination of the results. We consider an ATM-like system in which the cells pass through a tandem of nodes before arriving at the network exit buffer. We compare the analytic bounds suggested by our analysis with the actual results of a simulation program and find that: 1) the bounds, indeed, hold and 2) for practical reasons, the bounds are close enough to warrant very good prediction and to be used to design the size of the exit buffers.

A. Previous Work

The D+G/D/1 queue seems to be inherently different from either the D/G/1 queue (see, e.g. a study of that system with vacations in [7]), or the G/D/1 queue. A variation of the latter was studied in the context of ATM work in [8], where the arrival process into the queue is a Poisson cluster process and the model is the PCP/D/1 queue. Another variation was studied in [2], where the Geo/D/1 and the Geo/D/1/n queues were analyzed. The deterministically generated generally postponed arrival process introduced in our work inherently differs from those arrival processes in the fact that, despite its stochastic appearance to the queue, its generation is periodic and regular and, thus, its arrival rate is identical to the service rate in the system.

The problem of buffer design in ATM was addressed in [1], where a method for simulating a system to capture very low loss rates was presented,

II. The D+G/D/1 Queue: Model Formulation and Basic Relationships

We consider a discrete time model: time is slotted with the slot size equal to the processing time of a cell. The slot that starts at epoch t and ends at epoch $t+1$ is called *slot* t.

Cell arrival to the D+G/D/1 queue is governed by the following process. At the beginning of every time slot, exactly one cell is generated; let C_t denote the cell generated at time t. Upon generation, C_t does not enter the queue. Rather, C_t gets *postponed* and its queue entrance occurs at time $t+\mathcal{P}_t$, where $\mathcal{P}_t$ is a nonnegative integer-valued random variable (i.e., $\mathcal{P}_t$ gets its value from $0, 1, 2, \cdots$). $\mathcal{P}_t$ is called the *postponed value* of C_t; we assume that $\mathcal{P}_t$ has finite mean. We assume that the cells arrive at the queue just prior to the cell boundary, namely at $t^- = \lim_{\epsilon \to 0}\ t - \epsilon$.[1]

Services (cell processing) at the D+G/D/1 queue start exactly at slot boundary; thus, due to queue arrivals occurring at t^-, a cell that enters the queue at time t (more precisely t^-) is ready to be processed at slot t. The processing time (service time) of a cell is exactly one time unit.

A. Notation

Below, we assume that at $t=1$ the system starts operating with an empty queue. Cell C_t is said to be *postponed* at time τ if $t \leq \tau < t + \mathcal{P}_t$. In other words, C_t was generated on or before τ and entered the queue after τ. The collection of cells which are postponed at t is called the *postponed cell population at* t, and we let $P(t)$ denote the size of that population. The behavior of the arrival process and the notions of postpone values and postponed cell population are demonstrated in Fig. 2.

[1] Remark: Defining arrivals to occur at t^- is only for the purpose of organizing the events at the slot boundary. This could alternatively be done by having arrivals occur at t and all other events occurring at t^+. Our notation, therefore, does not preclude a postpone value $\mathcal{P}_t$ of value 0.

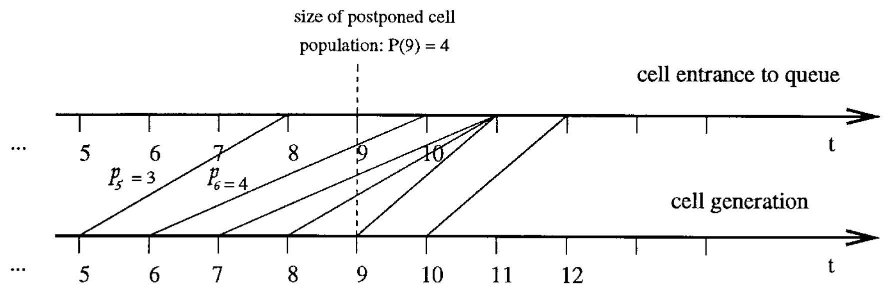


Fig. 2. The arrival process in the D+G/D/1 queue.

Let $A(t^-)$ (arrivals) denote the number of cells that enter the queue at time t^-, and $Q(t)$ denote the number of cells that are present at the buffer at time t (this includes the $A(t^-)$ arrivals).

Let $U(t) = P(t) + Q(t)$; note that $U(t)$ reflects the total unfinished work (backlog) present in the system at time t.

A *starvation* is an event in which no cell is processed due to the lack of cells ready to be processed. An epoch t is called a *starvation initiation point* if no cell processing starts at t; this happens if $Q(t) = 0$. Epoch $t+1$ is called, in this case, *starvation completion point*. Slot t is called, in this case, a *starvation slot*. Let $S(t)$ denote the total number of starvation initiation points which occur prior to epoch t. Thus $S(t) = \Sigma_{i=1}^{t-1} \mathbf{1}(Q(i) = 0)$ where $\mathbf{1}(\,)$ is the indicator function whose value is one if its argument is true and is zero otherwise.

B. Basic Relationships

The following is a basic relationship between the sizes of the postponed cell population $P(t)$ and the arriving cell population $A(t^-)$.

Theorem 1: For every $t > 1$

$$P(t) + A(t^-) = P(t-1) + 1. \tag{1}$$

Proof: The proof follows simply by observing that the population of postponed and arriving cells at t consists of the newly generated cell (at t) and the postponed cells of $t-1$.□

Next, we show that the expected size of the cell population, $E[P(t)]$, is bounded by that of the postpone distribution $E[\mathcal{P}]$.

Lemma 1: If the postpone values $\mathcal{P}_t$, $t = 1, 2, \cdots$, are identically distributed (like $\mathcal{P}$), and $E[\mathcal{P}]$ is finite, then the cell population distribution $P(t)$, $t = 1, 2, \cdots$, has finite mean, which obeys

$$E[P(t)] \le E[\mathcal{P}], \qquad t = 1, 2, \cdots \tag{2}$$

The proof of this lemma is given in the Appendix.

III. Basic Relationships for Infinite Buffer Systems: "Bad" and "Good" News

We start our analysis by examining the behavior of infinite buffer systems. The results derived here will be used later in the analysis of cell loss in finite buffer systems.

As can be seen below, the characteristics of the D+G/D/1 queue are quite different from those of common queueing systems.

The "bad news" of this section is that the system is unbounded, in the sense that when $t \to \infty$ the system's backlog is unbounded. The "good news" is that for *finite* values of t and some arbitrary value k (even relatively small values of k) the likelihood of reaching buffer occupancy k is very small.

The first lemma relates the number of starvation points to the total backlog work.

Lemma 2: For $t \ge 1$, $U(t) = S(t) + 1$.

Proof: The lemma follows from the fact that the number of cells generated in the slots $1, 2, \cdots, t-1$ is exactly $t-1$ and the number of cells processed during those slots is given by $t - 1 - S(t)$. The difference between these quantities is the number of cells generated in $1, 2, \cdots t-1$ but not processed in $1, 2, \cdots, t-1$. This quantity plus one (the cell generated at t) is the backlog at t, $U(t)$. Thus $U(t) = t-1-(t-1-S(t))+1 = S(t) + 1$. □

Next, we establish that the system is "instable." This instability is reflected in the fact that the system's backlog is monotonically nondecreasing.

Theorem 2: For $1 \le t < t'$, $U(t) \le U(t')$.

Proof: The proof follows directly from Lemma 2 and from the fact that $S(t)$ is monotonically nondecreasing (by definition). □

Unlike common queueing systems in which the growth of the unfinished work is typically gradual, the unfinished work in the D+G/D/1 queue is attributed to "arrival bursts." More precisely, a growth of the backlog to size k can occur only if a burst of k arrivals gets postponed concurrently. More precisely, the backlog can reach value k only if there exists an epoch $\tilde{t}$ in which the total number of postponed and arriving cells $P(\tilde{t}) + A(\tilde{t}^-)$ reaches k. This property is established in the next theorem.

Theorem 3: If $U(t_0) = k > 0$ then there exists an epoch $1 \le \tilde{t} \le t_0$ such that $A(\tilde{t}^-) + P(\tilde{t}) = k$.

Proof: Let $\tilde{t} = \min\{t | U(t) = k\}$. Obviously, $\tilde{t}$ must exist since $U(t_0) = k$; further, $\tilde{t} \le t_0$. For $k = 1$, we have $\tilde{t} = 1$, $U(\tilde{t}) = 1$, and $A(\tilde{t}^-) + P(\tilde{t}) = 1$ and the claim follows trivially.

For $k > 1$, we have $\tilde{t} > 1$ (since $U(1) = 1$) and we can examine the system at $\tilde{t} - 1$. Due to $\tilde{t}$ being minimal and due to the monotonicity of $U(\,)$ we have $U(\tilde{t}) = k$ and $U(\tilde{t} - 1) < k$. Thus, from Lemma 2, we have $S(\tilde{t}) = k - 1$ and $S(\tilde{t} - 1) < k - 1$. Therefore, $\tilde{t} - 1$ is a starvation initiation point and we must have $Q(\tilde{t} - 1) = 0$.

Now, since $Q(t)$ obeys $Q(t) = [Q(t-1) - 1]^+ + A(t^-)$, we must have $Q(\tilde{t}) = A(\tilde{t}^-)$. Thus $A(\tilde{t}^-) + P(\tilde{t}) = k$. □

An implication of Theorem 2, under quite a wide set of conditions (e.g., the postpone values are independent of each other), is that if the probability of reaching queue level k is positive then the probability of passing level k when $t \to \infty$ approaches 1, namely, $\lim_{t\to\infty} \Pr[U(t) \geq k] = 1$. This suggests a quite "unpleasant" behavior of the system. Nonetheless, in practice, this behavior is outweighed by Theorem 3, which can be used to bound the likelihood of reaching buffer level of k.

We later (Section V) use a stochastic postpone model to represent the system and will focus our efforts on using the property established in Theorem 3 to bound the probability of the system *entering* into large population state. The epochs at which the system enters such states are denoted k-*entrance epochs*. More precisely, an epoch t is called a k-*entrance epoch* if $A(t^-)+P(t) \geq k$. The importance of such an epoch is that if the system's backlog is lower than k prior to t, it increases to k (or above) at t.

The last result of this section bounds the amount of backlog work by the maximal postpone value. Recall that $\mathcal{P}_t$ is the postpone value of C_t.

Theorem 4: Let $k_{\max}$ be the maximal value of $\mathcal{P}_t$ for any value of t, namely $\mathcal{P}_t \leq k_{\max}$ for any t. Then: For every t, $U(t) \leq k_{\max}+1$.

Proof: The proof follows by observing at epoch t the joint population of $P(t)$ and $A(t^-)$. Of this joint population, the oldest cell must have been generated at $t-k_{\max}$ or later (due to the condition of the theorem), and the youngest cell must have been generated at t. Thus, due to the cells being distinct from each other we must have $P(t)+A(t^-) \leq k_{\max}+1$. Theorem 3 now implies that $U(t) \leq k_{\max}+1$. □

In situations in which $\mathcal{P}_t$ is unbounded, we cannot bound the buffer occupancy, which can reach any value k; in fact, due to Theorem 2, the probability of having k or more cells in the buffer, when $t \to \infty$ is 1. These cases are treated in Section V, in which we consider a stochastic model and provide bounds for the likelihood of reaching large population states.

IV. Analysis of Finite Buffer Systems

In this section, we analyze finite buffer systems. The model is equivalent to the one described earlier, but the buffer is assumed to be finite of size $B \geq 1$. For this reason, cell loss may occur in the system. Our interest is in deriving an upper bound for the probability of loss. This can be accomplished by relating the probability of loss to the probability of starvation (Theorem 5), and then establishing bounds on the probability of starvation (Section IV-C). We start with preliminary definitions.

A. System Modeling

Prior to the analysis of the system, assumptions on the system behavior must be made. We assume that events occur at the system at the following epochs (Fig. 3 depicts the events in the system).

1) Cell processing starts at the beginning of a slot and is triggered only if the number of cells contained at the queue is greater than zero $(Q(t)>0)$.

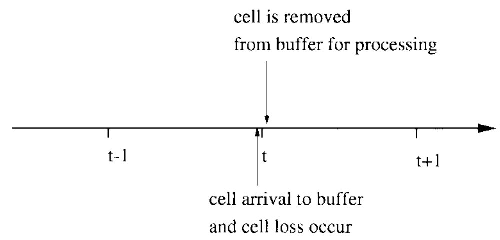


Fig. 3. Event sequence at slot boundary.

2) At the beginning of the slot, namely at t^+, the cell to be processed is removed from the buffer and the space it occupied is released. (During its processing, the processed cell is stored in a separate buffer that is not accounted for in our model.)

   Remark 1: Note that in our model, the space of queued cells and that of the processed cell are not shared. An alternative model is one in which those buffers are shared. A short analysis and the main results of this model are provided in [5].

3) Arrivals occur *during a slot*, and are accounted for as if they arrive at the end of the slot. We thus assume, as in the previous sections, that arrivals occur at t^- and let $A(t^-)$ denote their number.

4) Arrivals that find the queue full are *lost*. *Losses* occur at t^- and their number is denoted by $l(t^-)$. The number of losses obeys: $l(t^-)=[[Q(t-1)-1]^+ + A(t^-)-B]^+$. The actual arrivals that enter the queue at t is then given by $A(t^-)-l(t^-)$. The total number of losses occurring by time t is denoted $L(t)=\Sigma_{i=1}^{t} l(i^-)$.

Following these rules, the equations that govern the system behavior are as follows (recall that $Q(t)$ denotes the queue population at t)

$$l(t^-) = [[Q(t-1)-1]^+ + A(t^-) - B]^+ \tag{3}$$

$$Q(t) = [Q(t-1)-1]^+ + A(t^-) - l(t^-). \tag{4}$$

An equivalent equation for $Q(t)$ is

$$Q(t) = \min[[Q(t-1)-1]^+ + A(t^-), B]. \tag{5}$$

B. Basic Relationships between Starvation and Loss

The first relationship relates the losses $L(t)$ to the number of starvation events, $S(t)$.

Lemma 3: For a finite buffer system $L(t)+U(t)=S(t)+1$, for $t \geq 1$ where $U(t)=P(t)+Q(t)$, as defined (in Section II-A).

Proof: The lemma follows from the same logics leading to Lemma 2 (infinite buffer analysis). The number of cells not processed in the slots $1, 2\cdots, t-1$ is $S(t)$. At epoch t, all the cells not processed in the slots $1,\cdots,t-1$ must be either postponed, queued, or lost cells. Thus, accounting for the cell generated at t, we have $S(t)+1 = P(t)+Q(t)+L(t)$. □

Next, we relate the rate of loss, $\lim_{t\to\infty} E[L(t)]/t$ to the rate of starvation $\lim_{t\to\infty} E[S(t)]/t$.

Theorem 5: At steady state, the loss rate is equal to the starvation rate.

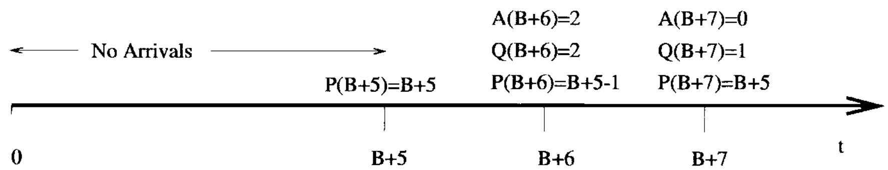


Fig. 4. Counter example for Theorem 6.

Proof: For a random variable X, let $E[X]$ be its expected value. From Lemma 3, we have that

$$\lim_{t\to\infty} \frac{E[L(t)]}{t} = \lim_{t\to\infty} \frac{E[S(t)]}{t} + \frac{1}{t} - \frac{E[U(t)]}{t} = \lim_{t\to\infty} \frac{E[S(t)]}{t}. \quad (6)$$

This is implied from the fact that $E[U(t)] = E[P(t)+Q(t)]$, and that $E[P(t)]$ is finite (see Lemma 1) and that $Q(t)$ is bounded from above by the buffer size B. □

C. Bounds on Work and Starvation

Next, we will establish an upper bound on $S(t)$, the number of starvation slots by epoch t. We will start by analyzing the system behavior using two lemmas. The first lemma deals with the *build-up* period, a period where the buffer first builds up. It starts at $t = 1$ and ends when, at the first time, $S(t) = B$. The second lemma deals with the *saturation period*, the period that starts when the first period ends.

Lemma 4: Let t_{S_i} denote the first epoch at which $S(t) = i$. Then for every $1 \le t < t_{S_B}$ the system behavior and the buffer contents are identical to those of the infinite buffer system (if it is subject to the same arrivals).

Lemma 5: For $t \ge t_{S_B}$, $U(t)$ obeys

$$B \le U(t) \le S(t) + 1.$$

The proofs of Lemma 4 and 5 are given in Appendix A.

Remark 2: Note that, unlike the infinite buffer system, in the finite buffer system, $U(t)$ does not grow monotonically since cells can be lost.

The next theorem analyzes the number of postponed cells ($P(t)$). Specifically, we characterize the epochs at which this function increases. This characteristic will then be used to establish bounds on the number of starvation points.

Prior to stating the theorem some more notation is required: An epoch t is called a P-*increment* point if $P(t) > P(t-1)$.

Theorem 6: Let $t \ge t_{S_B}$, then we have the following.

1) If t is a *starvation initiation* point, then:
   a) t is a P-*increment* point;
   b) $P(t) \ge B$.
2) If t is a P-*increment* point and $P(t) \ge B$, then t is not necessarily a *starvation initiation* point.

Proof:

1) If t is a *starvation initiation* point, then $Q(t) = 0$, implying that:
   a) $A(t^-) = 0$ [from (5)], and thus, from Theorem 1, t is a P-*increment* point;
   b) $P(t) = U(t)$, and thus, from Lemma 5, $P(t) \ge B$.
2) This is proved by providing a counter example. Such an example, depicted in Fig. 4, reflects a situation in which there are no arrivals at $0 \le t \le B+5$, two arrivals at $t = B+6$, and no arrivals at $t = B+7$. In that example, $t = B+7$ is a P-increment point but not a starvation point (since $Q(B+7) > 0$).

□

Corollary 1: The number of starvation points in the interval $[t_{S_B}, t]$ is bounded from above by the number of P-increment points for which $P(t) \ge B$.

Corollary 2: Assuming steady state, the starvation and loss rates are bounded by the size of the postponed cell population as follows:

$$\lim_{t\to\infty} \frac{E[L(t)]}{t} = \lim_{t\to\infty} \frac{E[S(t)]}{t} \le \lim_{t\to\infty} \Pr[P(t) \ge B]. \quad (7)$$

V. Datagram Networks: Systems with Independent Postpone Values

In this section, we consider systems in which different cells follow different paths through the network; thus, the delay incurred by one cell in the network can be assumed to be independent from that of other cells. This is modeled by having the postpone values of the different cells independent of each other.

Below, we assume that the postpone value of C_t, $\mathcal{P}_t$, is a random variable. We also assume that the postpone values are independent of each other, namely $\mathcal{P}_t$ and $\mathcal{P}_{t'}$ are independent of each other for every t and t', $t \ne t'$. Further, we assume that the postpone values are all taken from the same distribution, namely for every t, $\mathcal{P}_t$ is a random variable distributed like $\mathcal{P}$. Let $p_k = \Pr[\mathcal{P} = k]$ and let $P_k = \Pr[\mathcal{P} \ge k] = \Sigma_{i=k}^{\infty} p_i$, $k = 0, 1, 2, \cdots$.

Theorem 7: Assume that $\mathcal{P}_1, \mathcal{P}_2, \cdots$ are independent random variables, all distributed like $\mathcal{P}$. Then $\Pr[A(t^-) = k]$ is bounded as follows:

$$\lim_{t\to\infty} \Pr[A(t^-) = k] \le \sum_{i_1=0}^{\infty} p_{i_1} \sum_{i_2=i_1+1}^{\infty} p_{i_2} \cdots \sum_{i_k=i_{k-1}+1}^{\infty} p_{i_k}, \qquad k \ge 0. \quad (8)$$

Proof: Let $A_{i_1,i_2,\cdots,i_k}$ denote the event in which the cells $C_{t-i_1}, C_{t-i_2}, \cdots, C_{t-i_k}$ enter the queue at t. The probability of the event $A_{i_1,\cdots,i_k}$ is given by

$$\Pr[A_{i_1,\cdots,i_k}] = \overline{p}_0\overline{p}_1 \cdots \overline{p}_{i_1-1}p_{i_1}\overline{p}_{i_1+1} \cdots \overline{p}_{i_2-1}p_{i_2} \cdots p_{i_k} \quad (9)$$

where $\overline{x} = 1 - x$. Now, since for every j, $\overline{p}_j \leq 1$, this expression is bounded by

$$\Pr[A_{i_1,\cdots,i_k}] \leq p_{i_1} p_{i_2} \cdots p_{i_k}. \tag{10}$$

Finally, noting that

$$\Pr[A(t^-) = k] = \sum_{0 \leq i_1 \leq i_2 \cdots \leq i_k \leq \infty} \Pr[A_{i_1,\cdots,i_k}] \tag{11}$$

the theorem is proved. □

A similar bound can be derived regarding the nonqueued backlog (postponed cells plus queue arrivals) at t.

Theorem 8: The density of the nonqueued backlog distribution $\Pr[P(t) + A(t^-) = k]$ is bounded as follows:

$$\lim_{t\to\infty} \Pr[P(t) + A(t^-) = k] \leq 1 \cdot \sum_{i_2=1}^{\infty} P_{i_2} \sum_{i_3=i_2+1}^{\infty} P_{i_3} \cdots \sum_{i_k=i_{k-1}+1}^{\infty}, \ P_{i_k} \geq 1. \tag{12}$$

Proof: The proof goes along similar lines to that of Theorem 7 and by noting that: 1) the probability that cell C_{t-x} is either postponed or arriving at t is given by P_x and 2) the cell located at $i_1 = 0$ is always postponed or arriving at $t = 0$. □

Remark 3: The bounds given in (8) and (12) increase somewhat if $\mathcal{P}_t$ is known to be $\mathcal{P}_t > 0$ (which is the case in many applications) since then C_t is always postponed at t and C_{t-1} is always postponed or arriving at t.

A. Exact Analysis for Geometric Postpone Distribution

Below, we derive the steady state distributions assuming that $\mathcal{P}_1, \mathcal{P}_2, \cdots$, are independent random variables, all distributed geometrically, i.e., $p_k = \Pr[\mathcal{P} = k] = \overline{p}p^k$ and therefore $P_k = \Pr[\mathcal{P} \geq k] = p^k$.

First, we provide exact expressions for the postpone distribution $P(t)$. Let $\pi_i = \lim_{t\to\infty} \Pr[P(t) = i]$ and $\Pi(z) = \Sigma_{i=0}^{\infty} \pi_i z^i$. From the evolution of the number of postponed cells, we have

$$\begin{aligned} \pi_0 &= \sum_{i=0}^{\infty} \pi_i \overline{p}^{i+1} \\ \pi_j &= \pi_{j-1} p^j + \sum_{i=j}^{\infty} \pi_i \binom{i+1}{j} \overline{p}^{i-j+1} p^j, \qquad j \geq 1. \end{aligned} \tag{13}$$

Multiplying (14) by z^j and summing for $j \geq 1$, we obtain

$$\begin{aligned} \sum_{j=1}^{\infty} \pi_j z^j &= \sum_{j=1}^{\infty} \pi_{j-1} (pz)^j \\ &\quad + \sum_{j=1}^{\infty} \sum_{i=j}^{\infty} \pi_i \binom{i+1}{j} \overline{p}^{i-j+1} (pz)^j \end{aligned} \tag{14}$$

which after simple algebra yields

$$\Pi(z) = (\overline{p} + pz)\Pi(\overline{p} + pz). \tag{15}$$

Using (15) recursively, we have

$$\Pi(z) = \prod_{i=1}^{\infty} [1 - p^i(1 - z)]. \tag{16}$$

By taking the kth derivative $(k = 0, 1, 2, \cdots)$ of (16) with respect to z and substituting $z = 0$, we obtain

$$\begin{aligned} \pi_0 &= \prod_{i=1}^{\infty} (1 - p^i) \\ \pi_k &= \pi_0 \sum_{i_1=1}^{\infty} \alpha_{i_1} \sum_{i_2=i_1+1}^{\infty} \alpha_{i_2} \cdots \sum_{i_k=i_{k-1}+1}^{\infty} \alpha_{i_k} \\ &\qquad k \geq 1. \end{aligned} \tag{17}$$

where $\alpha_i = p^i/(1 - p^i)$.

Corollary 3: Using Theorem 1, the expressions derived in (17) and in (16) can be used directly to derive the distribution of $q_i = \lim_{t\to\infty} \Pr[P(t) + A(t^-) = i]$. Specifically, we have $q_0 = 0$ and $q_i = \pi_{i-1}$ for $i \geq 1$.

Lastly, we provide closed form expressions for upper bounds on the distribution of $A(t^-)$ and $P(t) + A(t^-)$. These are based on the bounds derived in (8) and (12). Under the assumption of geometric distribution of the postpone values, these bounds become

$$\lim_{t\to\infty} \Pr[A(t^-) = k] \leq \frac{\overline{p}^k}{p^k} \prod_{i=1}^{k} \frac{p^i}{1 - p^i}, \qquad k \geq 0 \tag{18}$$

$$\Pr[P(t) + A(t^-) = k] \leq \prod_{i=1}^{k-1} \frac{p^i}{1 - p^i}, \qquad k \geq 1. \tag{19}$$

Remark 4: Note that for every value of $0 < p < 1$ and $k > 1$ the right hand side of (19) is smaller than that of (18). Note also that the bound in (19) can also be derived by using Theorem 1 and (18) and (17).

Corollary 4: From Theorem 1 and (19), we get

$$\lim_{t\to\infty} \Pr[P(t) = k] \leq \prod_{i=1}^{k} \frac{p^i}{1 - p^i}, \qquad k \geq 0. \tag{20}$$

From (20), we may now derive expressions for the probability of a p-increment point. This is given by $\Pr[P(t) = k, P(t + 1) = k + 1]$ and can be computed as follows:

$$\begin{aligned} \Pr[P(t+1) = k+1, P(t) = k] &= \Pr[P(t+1) = k+1 | P(t) = k] \cdot \Pr[P(t) = k] \\ &\leq p^{k+1} \prod_{i=1}^{k} \frac{p^i}{1 - p^i}, \qquad k \geq 0 \end{aligned} \tag{21}$$

where the inequality results from (20).

The practical use of these inequalities is as follows. Equation (19) can be used in conjunction with the analysis of infinite buffer systems to bound the likelihood of k-entrance epochs. Equation (22) can be used in conjunction with the analysis of the finite buffer system (Corollary 1) to bound the likelihood of loss (and starvation) as follows:

$$\lim_{t\to\infty} \frac{E[S(t)]}{t} \leq \sum_{k=B}^{\infty} p^{k+1} \prod_{i=1}^{k} \frac{p^i}{1 - p^i}. \tag{22}$$

VI. Virtual Circuit Networks: Systems with Tandem-Like Dependent Postpone Values

In this section, we consider a system in which all cells of a session follow the same path. This implies that the stream of cells under consideration follows a tandem of queues. This fact, combined with an assumption that the queues use the FIFO strategy, implies direct dependencies between the delays incurred by different cells through the network. In particular, cell i cannot overtake cell $i-1$. This is expressed in the next proposition.

Proposition 9: For $k = 0, 1, \cdots$, if $\mathcal{P}_t = k$ then $\mathcal{P}_{t+1} \geq k-1$.

Next we establish two simple conditions for cell accumulation at the end buffer:

Theorem 10: The size of the postpone population at time t depends on the postpone values of past cells as follows.

1) $P(t) = k+1$ if, and only if, $\mathcal{P}_{t-k} > k$ and $\mathcal{P}_{t-k-1} \leq k+1$.
2) $P(t) \geq k$ if, and only if, $\mathcal{P}_{t-k+1} \geq k$.

Proof: The proof follows directly from the proposition.

Theorem 10 leads to a striking identity between the distribution of postponed values $\mathcal{P}_t$ and the distribution of number of postponed cells at certain moments, $P(t)$;

Theorem 11: The limiting distribution of the number of cells postponed in the system is identical to the limiting distribution of the postpone values, namely

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}[P(t) = k] = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}[\mathcal{P}_t = k] \tag{23}$$

where $\mathbf{1}(\,)$ is the indicator function.

Proof: From Theorem 10 (2) and Proposition 9, it is easy to see that for any value of T and k, we have

$$k \geq \sum_{t=1}^{T} \mathbf{1}[\mathcal{P}_t \geq k] - \sum_{t=1}^{T} \mathbf{1}[P(t) \geq k] \geq 0. \tag{24}$$

Applying this inequality to k and $k-1$, and subtracting these two equations from each other, we get

$$k \geq \sum_{t=1}^{T} \mathbf{1}[\mathcal{P}_t = k-1] - \sum_{t=1}^{T} \mathbf{1}[P(t) = k-1] \geq -(k-1). \tag{25}$$

Finally, dividing by T and taking T to infinity yields the proof. □

Remark 5: Note that Theorem 11 is in fact a generalization of Little's law [6]. The original law, which applies to a very wide collection of systems, relates the expected delay incurred by customer (W) to the arrival rate (λ) and the expected number of customers residing in the system (L) by the formula $L = \lambda W$. A straightforward application of Little's law to the population of postponed cells yields $E[P] = 1 \cdot E[\mathcal{P}]$, where $P = \lim_{t\to\infty} P(t)$, and 1 is the arrival rate into the system. However, as shown by Theorem 11, a much stronger law applies in our case: The relation between the population size, the delay, and the arrival rates holds not only to their **expected values** but also to their **full distribution**, namely $\Pr[P = k] = \Pr[\mathcal{P} = k]$.

Note that the distributional form of Little's law was studied in [3] and [4], but in the framework of Poisson arrivals. Here, this distributional form of the law is observed in the context of periodic (regular) arrivals.

To conclude this section, we note that Theorem 11 And Corollary 2 yield a computable bound on the loss rate and starvation rate.

VII. Practical Use

The analysis provided above (Corollary 2, Theorem 1 and Theorem 8 for independent postpone values with general distributions, (22) for independent postpone values with geometric distributions, and Corollary 2 and Theorem 3 for tandem-like dependent postpone values) yields bounds on loss and starvation expressed in terms of the postpone distribution $\mathcal{P}_t$.

For actual networks, the postpone values are the delays incurred in the networks. These are either known, or procedures for their derivation (exact or approximate) are typically available.

Thus, the derivation of the loss rate at the end buffer should be conducted as follows: 1) derive, by other means, the delays incurred in the network and use them as the postpone values in our model and 2) use our analysis to derive the bounds on the loss rate as a function of the buffer size (and the postpone values).

VIII. Simulation Results

The aim of this section is to use simulation to examine, in the context of ATM, the quality of the loss probability analytic bounds derived in the paper. We examine three bounds. The first bound, denoted P_{inc}, reflects the bound established in corollary 1. P_{inc} is defined as

$$P_{\text{inc}} = \Pr[P(t) \geq P(t-1), P(t) \geq B]. \tag{26}$$

Since, by Theorem 5, the probability of loss is equal to that of starvation, $\mathcal{P}_{\text{inc}}$ is a bound on the probability of loss.

The second bound is $\Pr[P(t) \geq B]$ which is, by definition, an upper bound on P_{inc}.

The third bound is $\Pr[\mathcal{P}_t \geq B]$ which is identical, by Theorem 11, to $\Pr[P(t) \geq B]$.

Note that the first two bounds are, in principle, hard to compute, and thus will be evaluated via the simulation. In contrast, the third bound reflects the postpone distribution which depends only on the delay distribution incurred by a cell while passing through the network. This measure can be derived by other means, and is carried out in this example by analytic means.

A. Simulation Description

The system simulated is a specific application of the data telecommunication session analyzed above. As mentioned, a session consists of a data source node and a destination node whose processing/transmission speeds are identical to each other and which are connected by a network whose speed and delay can be arbitrary. For the simulation, we chose a specific network consisting of four tandem switches, each with 16 inputs and outputs. Each of the 16 outputs has a queue for the data destined for that output from all 16 inputs. A cell traversing the network will accumulate delays at the queues of the four switches. The total delay is eventually translated to the cell's postponed value at the network's exit buffer.

The transmission speed of the data in the network is greater than the transmission speeds at the source and destination nodes. The parameter that determines the ratio of these speeds is denoted the *rate ratio*. The cell delay accumulated at the four switches in the network must be translated to the speed in the destination node by dividing the delay by the rate-ratio parameter. This value is the postponed value (denoted through the paper as $\mathcal{P}_t$) for the network exit buffer. As explained in the paper, a cell generated at time t gets postponed and its queue entrance at the network's exit buffer occurs at time $t + \mathcal{P}_t$.

In order to simplify the system, one of the 16 inputs to each of the switches will transmit only the cells of the session being simulated. The session cells enter the first switch according to a constant speed determined by the rate-ratio parameter; then, their entrance to switch i is determined by their entrance to switch $i-1$ and the delay incurred in switch $i-1$. The other 15 switch inputs supply the rest of the data entering the switch, each following a Bernoulli process in a stochastic manner, so that the total data entering the buffer at the switch obeys the set load parameter.

The simulation has four input parameters, load, rate-ratio, duration, and buffer size, and measures six others: 1) loss; 2) number of starvation events; 3) P-increment points; 4) $Q(t)$; 5) $P(t)$; and 6) $\mathcal{P}_t$ (all defined above).

B. Computation of Analytic Prediction

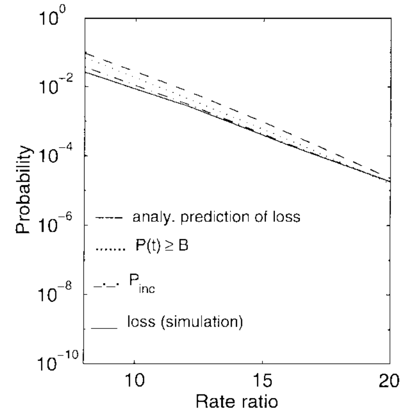
The postpone value $\mathcal{P}_t$ in the D+G/D/1 queue model is equal to the delay incurred by a cell while passing through the network. This is given by

$$\mathcal{P}_t = \left\lfloor \frac{\sum_{i=1}^{4} d_t^i}{rate\text{-}ratio} \right\rfloor \tag{27}$$

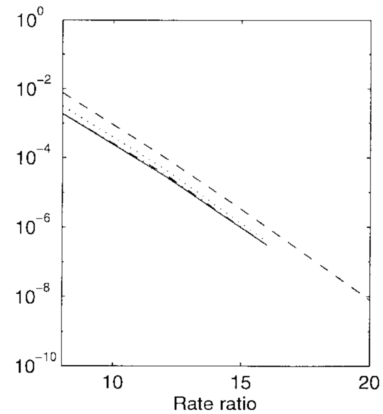
where d_t^i is the delay incurred by the tth cell at switch i.

Thus, a convolution of the delays incurred in the individual switches is used to obtain the analytic prediction of $\mathcal{P}_t$. This would be *precise* if the switch delays were independent of each other. In our case, one may assume that these are close to being independent due to the extent of traffic mixing occurring in the switches. However, these are not *exactly independent* due to the tandem-like structure.

The delay distribution at switch i, d_t^i, is computed numerically from the switch arrival process. This is assumed to consist of the sum of 16 independent Bernoulli processes, each with parameter p =load/16.



(a)



(b)

Fig. 5. Cell-loss rate as a function of the network/source rate ratio (Load $= 0.8$). (a) $B = 2$. (b) $B = 3$.

C. Results

Our results are exhibited in four figures. Figs. 5–7 depict the loss rate as a function of the rate-ratio, for different buffer sizes (B) and loads. The figures contain each four curves: the actual loss rate incurred in the simulation (solid line), and the three bounds P_{inc}, $\Pr[P(t) \geq B]$, and the analytic prediction of loss (and $\mathcal{P}_t$). While the first two bounds are measured by the simulation, the third one is computed analytically. The actual values of the quantity $\Pr[\mathcal{P}_t \geq B]$ were measured in the simulation as well, and were found to be exactly equal to $\Pr[p(t) \geq B]$.

Fig. 8 depicts the loss rate as obtained by the simulation (solid line), and the analytic prediction (dashed line) as functions of the buffer size, for different values of load and rate ratio.

D. Discussion of the Results

The results suggest that the effect of the buffer size on the cell-loss rate is very dramatic. Fig. 5, for example, demonstrates that under approximately realistic parameters (source/network rate ratio of 20 and load of 0.8), the cell-loss rate drops more than three orders of magnitude (from more than 10^{-5} to 10^{-8}). Thus, the indication given in the paper

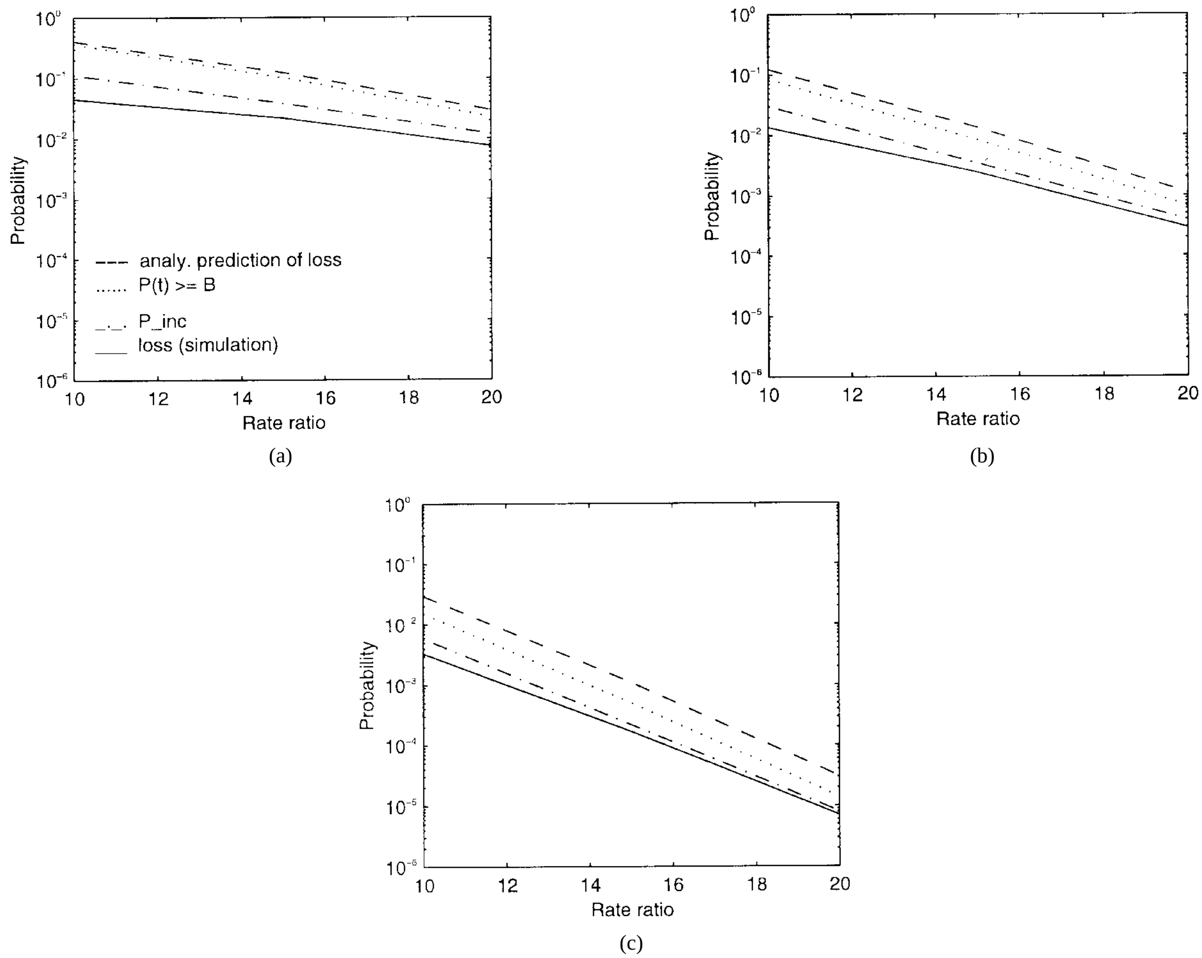


Fig. 6. Cell-loss rate as a function of the network/source rate ratio (Load $= 0.9$). (a) $B = 2$. (b) $B = 3$. (c) $B = 4$.

that likelihood of loss rapidly diminishes with the buffer size is verified by the simulation results.

Figs. 5–7 demonstrate that on a log-scale the cell loss rate decreases close to linearly with the rate ratio. This suggests that the rate-ratio curves can be used as good predictors (by the way of extrapolation) for the derivation of measures that are hard to simulate (very low cell loss rates).

Fig. 8 exhibits clearly the effectiveness of the bounds for different load values and rate-ratios values. We observe that the quality of the bound (as an approximation and predictor) improves with the rate ratio and degrades with the load. Since we are interested mainly in rate-ratio values in the area of 30 and load values of approximately 0.8, we have a tight bound.

It is important to mention that it is impossible (time wise) to produce simulation results for loss probabilities in the range lower than 10^{-08}, but the prediction (bound) for these parameter values can be easily computed.

APPENDIX
PROOFS OF LEMMAS 1, 4 AND 5

Proof of Lemma 1: For $t \geq 1$, $P(t)$ is the number of cells generated prior to t but which enter the queue after t. Thus, we have

$$P(t) = \sum_{i=0}^{t-1} \mathbf{1}(\mathcal{P}_{t-i} > i) \tag{28}$$

where $\mathbf{1}(\,)$ is the indicator function. Thus, we have

$$\begin{aligned} E[P(t)] &= \sum_{i=0}^{t-1} E[\mathbf{1}(\mathcal{P}_{t-i} > i)] = \sum_{i=0}^{t-1} \Pr[\mathcal{P}_{t-i} > i] \\ &= \sum_{i=0}^{t-1} \Pr[\mathcal{P} > i] = \sum_{i=1}^{t} \Pr[\mathcal{P} \geq i] \\ &\leq \sum_{i=1}^{\infty} \Pr[\mathcal{P} \geq i] = E[\mathcal{P}] \end{aligned} \tag{29}$$

where the first equality stems from the fact that the expected value of a sum is the sum of the expected values and the last equality stems from the fact that $\mathcal{P}$ is a nonnegative discrete random variable. □

Proof of Lemma 4: Let the first epoch when loss occurs be denoted by t^{-}_{loss}. The system is identical to an infinite system if there is no loss in the system, namely if $t^{-}_{\text{loss}} \geq t_{S_B}$, which we prove next.

For the contradiction, assume that $1 \leq t^{-}_{\text{loss}} < t_{S_B}$.

From Lemma 3 and the definition of t_{S_B}, we have

$$L(t_{\text{loss}}) + U(t_{\text{loss}}) = S(t_{\text{loss}}) + 1 < B + 1. \tag{30}$$

On the other hand if loss occurs at t^{-}_{loss} then $L(t_{\text{loss}}) \geq 1$ and $U(t_{\text{loss}}) \geq B$ (since loss occurs when $Q(t) = B$). Hence, $L(t_{\text{loss}}) + U(t_{\text{loss}}) \geq B + 1$, which contradicts (30). Thus

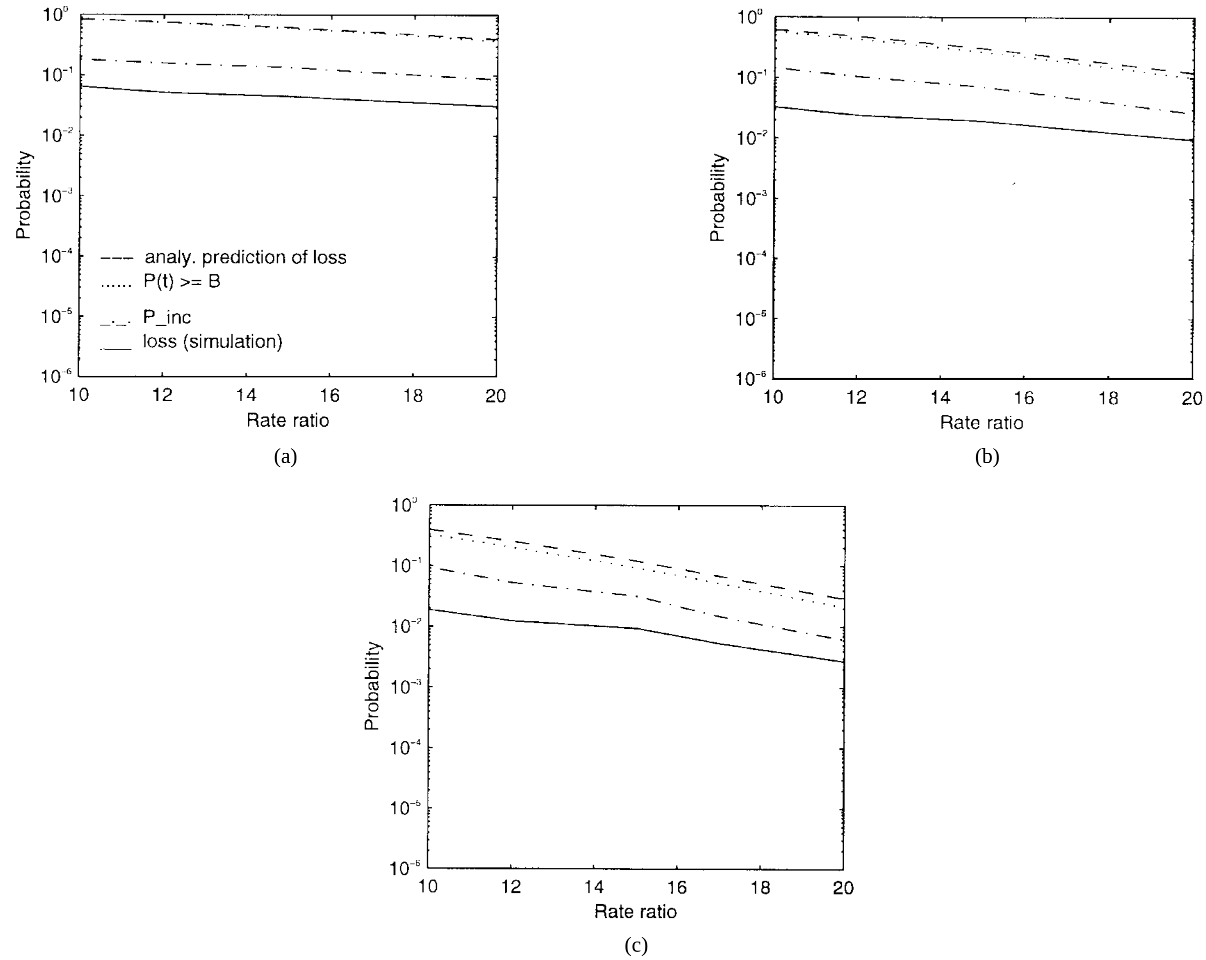


Fig. 7. Cell-loss rate as a function of the network/source rate ratio (Load $= 0.95$). (a) $B = 2$. (b) $B = 3$. (c) $B = 4$.

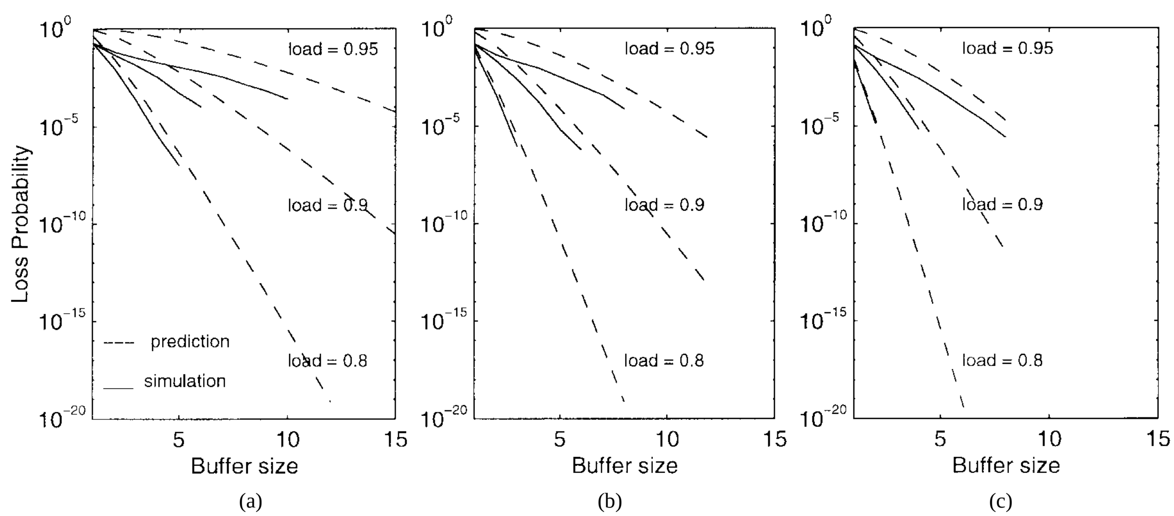


Fig. 8. Cell-loss rate as a function of the buffer size. (a) Rate ratio = 10. (b) Rate ratio = 15. (c) Rate ratio = 20.

by the contradiction for $1 \le t < t_{S_B}$ no loss occurs, and the system is identical to an infinite buffer system. □

Proof of Lemma 5: The proof of $U(t) \le S(t) + 1$ follows trivially from Lemma 3.

To prove $B \le U(t)$, let us first consider $t = t_{S_B}$. From Lemma 3 and the definition of t_{S_B}, we have $L(t_{S_B}) + U(t_{S_B}) = S(t_{S_B}) + 1 = B + 1$. We prove that $L(t_{S_B}) \le 1$ and thus $U(t_{S_B}) \ge B$.

For $t < t_{S_B}$ there is no loss (Lemma 4), hence $L(t_{S_B}) = l({t_{S_B}}^-)$. Now t_{S_B} is a *starvation completion* point, and hence

$$\begin{aligned} l(t_{S_B}) &= [A({t_{S_B}}^-) - B]^+ \le [P(t_{S_B} - 1) + 1 - B]^+ \\ &\le [U(t_{S_B} - 1) + 1 - B]^+ \\ &= [S(t_{S_B} - 1) + 1 + 1 - B]^+ = 1. \end{aligned} \tag{31}$$

(using (3), Theorem 1, Lemma 3, and the definition of t_{S_B}.) Thus, $l({t_{S_B}}^-) \le 1$ and $U(t_{S_B}) \ge B$.

Next, we will show by induction that for $t > t_{S_B}$ if $U(t) \geq B$ then $U(t+1) \geq B$.

From (4) and Theorem 1, it follows that

$$\begin{aligned} U(t+1) &= ([Q(t)-1]^+ + A(t^-+1) - l(t^-+1)) \\ &\quad + (P(t) + 1 - A(t^-+1)) \\ &\geq Q(t) - l(t^-+1) + P(t) \\ &\geq B - l(t^-+1). \end{aligned} \tag{32}$$

(where the first inequality results from algebraic manipulation and the second from the inductive assumption).

Now if $l(t^-+1) = 0$, it follows that $U(t+1) \geq B$. And, if $l(t^-+1) > 0$, the queue is full $(Q(t+1) = B)$, and thus $U(t+1) \geq B$, which completes the proof. □

References

[1] F. Bernabei, R. Ferretti, M. Listanti, and G. Zingrillo, "ATM system buffer design under very low cell loss probability constraints," in *Proc. INFOCOM'91*, pp. 929–938.

[2] A. Gravey, J. R. Louvion, and P. Boyer, "On the Geo/D/1 and Geo/D/1/n queues," *Perform. Eval.*, vol. 11, no. 2, pp. 117–125, 1990.

[3] J. Keilson and L. D. Servi, "A distributional form of Little's law," *Oper. Res. Lett.*, vol. 7, pp. 223–227, Oct. 1988.

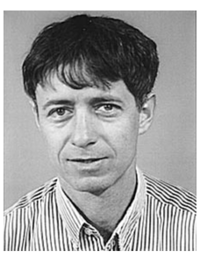
[4] ——, "The distributional form of Little's law and the Fuhrmann-Cooper decomposition," *Oper. Res. Lett.*, vol. 9, no. 4, pp. 239–247, 1990.

[5] H. Levy, T. Mendelson, M. Sidi, and J. Keren-Zvi, "Sizing exit buffers in ATM networks," Transport Division, Tadiran (ECI) Telecommunications, Petach-Tikva, Israel, Tech. Rep., June 1997.

[6] J. D. C. Little, "A proof of the queueing formula: $L = \lambda W$," *Oper. Res.*, vol. 9, pp. 383–387, 1961.

[7] L. D. Servi, "D/G/1 queues with vacations," *Oper. Res.*, vol. 34, pp. 619–629, 1986.

[8] K. Sohraby, "Delay analysis of a single server queue with poisson cluster arrival process arising in ATM networks," in *Proc. GLOBECOM'89*, pp. 611–616.

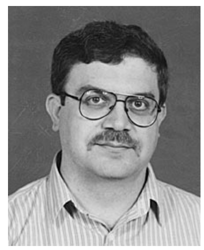


Hanoch Levy (S'83–M'87) received the B.A. degree in computer science (with distinctions) from the Technion–Israel Institute of Technology, Haifa, Israel, in 1980, and the M.Sc. and Ph.D. degrees in computer science from the University of California at Los Angeles in 1982 and 1984, respectively.

From 1984 to 1987 he was a Member of Technical Staff in the Department of Teletraffic Theory, AT&T Bell Laboratories. In 1987 he joined the Department of Computer Science, Tel-Aviv University, Tel-Aviv, Israel. During the years 1992–1996, he was with the School of Business and RUTCOR, Rutgers University, New Brunswick, NJ, on leave from Tel-Aviv University. Currently, he serves as the Chairman of the Department of Computer Science, Tel-Aviv University. His interests include computer communications networks, QoS in high-speed networks, performance evaluation of computer systems, and queueing theory.

Tzippi Mendelson received the B.Sc. degree in theoretical physics and computer science from Bar-Ilan University, Ramat Gan, Israel, in 1994, and the M.Sc. degree in theoretical physics from the Hebrew University, Jerusalem, in 1999.

From 1996 to 1998, she was with ECI Telecom, Petach-Tivka, Israel, researching ATM networking. During 1999, she has been teaching computer science and mathematics at the Open University, Tel-Aviv, Israel, and Lipshitz College, Jerusalem, Israel.



Moshe Sidi (S'77–M'78–SM'87) received the B.Sc., M.Sc., and D.Sc. degrees from the Technion–Israel Institute of Technology, Haifa, Israel, in 1975, 1979, and 1982, respectively, all in electrical engineering.

In 1982, he joined the Electrical Engineering Department, the Technion. During 1983–1984, he was a post-doctoral Associate in the Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology, Cambridge, MA. During 1986–1987, he was a Visiting Scientist with IBM, T.J. Watson Research Center, Yorktown Heights, NY. He currently serves as Head of the Center for Communications and Information Technologies, the Technion. He is co-author of the book *Multiple Access Protocols: Performance and Analysis* (New York: Springer-Verlag, 1990). His research interests are in queueing, modeling, and performance evaluation of computer communication networks, traffic characterization, and guaranteed grade of service in high-speed networks and wireless networks, and multiple access protocols.

Dr. Sidi served as the Editor of Communication Networks for the IEEE TRANSACTIONS ON COMMUNICATIONS from 1989 to 1993, Associate Editor of Communication Networks and Computer Networks for the IEEE TRANSACTIONS ON INFORMATION THEORY from 1991 to 1994, and as an Editor for the IEEE/ACM TRANSACTIONS ON NETWORKING from 1993 to 1997. Currently, he serves as an Editor for the *Wireless Journal*.

Joseph Keren-Zvi (A'92) received the B.Sc and M.Sc. degrees in electrical engineering from the University of Houston, TX, in 1974 and 1977, respectively.

From 1975 to 1985, he was an Engineer with both Texas Instruments Incorporated and Motorola, working on communication systems. From 1985 to 1992, he was a Consulting Engineer in the research group of CODEX, Canton, MA, working on fast packet technology. In 1992, he joined ECI Telecom, Petach-Tivka, Israel, as a System Engineer, working on an ATM transport systems. Since 1997, he has been a Chief Engineer in Marketing, working on multi-layer bandwidth management.