The Superposition of Discrete-Time Markov Renewal Processes with an Application to Statistical Multiplexing of Bursty Traffic Sources

K. M. Elsayed
H. G. Perros

The Superposition of Discrete-Time Markov Renewal Processes with an Application to Statistical Multiplexing of Bursty Traffic Sources *

Khaled M. Elsayed   Harry G. Perros

Department of Computer Science

and

Center for Communications and Signal Processing

North Carolina State University

Raleigh, NC 27695-8207

Abstract

In this paper we introduce a methodology for approximately characterizing the superposition process of $N \geq 2$ arbitrary discrete-time Markov Renewal Processes (MRP). The superposition process is also a MRP with a state space that grows exponentially with N. We introduce the arbitrary on/off traffic source model as a special case of the general MRP. For this special case, we devise an iterative algorithm which can be used to characterize the superposition process in a compact form. Subsequently, a queueing model for a FIFO finite-buffer multiplexer with arbitrary on/off input sources is analyzed. We provide extensive numerical and validation results for the algorithms introduced in the paper. We also study the effect of some of the statistical properties of on/off input sources on the multiplexer's performance.

1 Introduction

In this paper, we consider the problem of characterizing the superposition of multiple discrete-time Markov Renewal Processes (MRP). A lot of research has been reported in

1

the area of characterizing the superposition process of renewal and Markovian processes. However, to the best of our knowledge characterizing the superposition of MRP's has not as yet been adequately addressed.

Non-Markovian processes are more versatile since they can provide realistic models of arrival processes in an ATM environment. However, the analysis of queueing systems with non-Markovian input is considerably more complex than that with renewal and Markovian arrival processes.

Cherry and Disney [1] studied the superposition of two continuous-time MRP's. The structure of the interval process resulting from superposing two independent MRP was characterized. The resulting stochastic process has a very large number of states which limits the applicability of the model to two processes.

Korolyuk [6] introduced a mathematical model for the superposition of multiple independent continuous-time MRP's. The model keeps track of the time each process spends in the current state. This limits the model's applicability to cases where analytic expressions for the sojourn times between states can be found. The author also introduced the notion of *phase space lumping,* an aggregation process, in order to reduce dimensionality of the MRP.

Following a similar methodology to the one presented in [1] and [6], we characterize the superposition of $N \geq 2$ independent discrete-time MRP's. The constructed superposition is a MRP with a state space that grows exponentially with N. Only small values of N can be handled on a conventional computer. An aggregation method was developed in the case of on/off processes, where the on and off periods have an arbitrary distribution. Such a process is hereafter referred to as an arbitrary on/off process. This aggregation causes a distortion of the statistical properties of the original superposition. However, the aggregation works well in the case of Interrupted Bernoulli Processes. Finally, a FIFO finite buffer statistical multiplexer with N arbitrary on/off arrival processes was analyzed. The analysis is non-standard due to the new model of the superposition of the arrival processes.

Sohraby [8] presented innovative results on the tail behavior of a multiplexer with infinite waiting room shared by multiple arbitrary on/off sources. For the case of hetero-

geneous sources, the model is valid only in heavy traffic. For the case of homogeneous sources, the model can be used for all levels of multiplexer utilization. In the heterogeneous case, the asymptotic decay parameter is a simple function of the first two moments of the on and off periods and peak rates. However, our results show that for the case of a multiplexer with finite buffer, more parameters are needed than just the first two moments of the on and off period lengths (when peak rate is equal to link speed).

The rest of this paper is organized as follows. In section 2, we first provide a quick review of Markov renewal processes and present some results from discrete-time renewal theory. We then proceed to describe the methodology for characterizing the superposition of multiple independent MRP's. In section 3, we present the arbitrary on/off source model. A step-wise algorithm which uses aggregation in order to obtain a superposition process of arbitrary on/off sources with a manageable number of states is introduced in section 3.2. In section 4, a finite buffer FIFO statistical multiplexer shared by multiple arbitrary on/off traffic sources is introduced and analyzed. In section 5, we present extensive validation results of the algorithms presented in this paper. In section 6, we examine the effects of some of the parameters of an arbitrary on/off source on the performance of a statistical multiplexer. Section 7 gives the conclusions of this paper.

2   The Superposition of Multiple Independent Discrete-time Markov Renewal Processes

2.1   A Review of Discrete-time Markov Renewal Processes and Some Basic Results from Renewal Theory

A Markov renewal process (MRP) is a stochastic process which moves from one state to another with a random sojourn times which has a distribution that depends on the state being visited as well as the next state to be entered. The successive states visited by the MRP form a Markov chain. Let Ξ be the state space of an MRP.

Definition: The stochastic process $(X, T) = \{X_r, T_r; r \in Z^+\}$ where $X_r \in \Xi$, $T_r = n\tau$,

3

$n \in Z^+$, and $0 = T_0 \leq T_1 \leq T_2 \cdots$, is a discrete-time MRP with state space Ξ if

$$Pr\left\{X_{r+1} = x, T_{r+1} - T_r = k\tau | X_0, \cdots, X_r; T_0, \cdots, T_r\right\} = Pr\left\{X_{r+1} = x, T_{r+1} - T_r = k\tau | X_r\right\}$$

for all $r, k \in Z^+$ and $x \in \Xi$. Let us further assume that (X, T) is time-homogeneous; that is $P\{X_{n+1} = y, T_{n+1} - T_n = k | X_n = x\} = G(x, y, k)$ independent of n. The family of probabilities $G = \{G(x, y, k) : x, y \in \Xi, k \in Z^+\}$ is called a discrete-time semi-Markov kernel on Ξ. The process (X) is called the associated semi-Markov process (SMP) of the MRP (X, T). Whenever appropriate, we would use the term SMP in place of MRP.

The sum $p(x, y) = \sum_{k \geq 0} G(x, y, k)$ is not necessarily equal to one, but $p(x, y) \geq 0$ and $\sum_{y \in \Xi} p(x, y)$ must be equal to one. The $p(x, y)$ are in fact the transition probabilities for some Markov Chain with state space Ξ and probability transition matrix $P = [p(x, y)]$.

We now review some basic results from renewal theory that will be used below to construct the superposition process. Let $f(k)$, $0 \leq f(k) \leq 1$, $k \geq 0$, be a probability mass function. Let $\omega = \sum_{k=0}^{\infty} f(k)$, $0 < \omega \leq 1$, and $\bar{M} = \sum_{k=0}^{\infty} kf(k)$. Let $F(k) = \sum_{l=0}^{k} f(l)$, be the associated cumulative probability density function of length up to k. The probability mass function of the residual life-time is given by :

$$\hat{f}(k) = \omega\left[\frac{\omega - F(k-1)}{\bar{M}}\right], \quad k = 1, 2, \cdots \tag{1}$$

Note that when ω is equal to 1, we get the known results for discrete-time renewal theory [9].

2.2 Characterization of The Superposition Process of Multiple Independent Markov Renewal Processes

Consider $N \geq 2$ independent discrete-time MRP's. Each individual MRP i is characterized in terms of a semi-Markov kernel $G_i = [g_i(x, y, k)]$ defined over the set of states $1, 2, \cdots N_i$, $N_i \geq 1$. In order to characterize the superposition process, we have to define the states of the process and then for each state find the distribution of the sojourn time between the state and any other directly accessible state of the superposition process.

In [1] the superposition state descriptor is a vector consisting of the current state of each component process and the time each process has spent in its current state since its last transition. It is clear that such a characterization results in an enormous state space, since the time a component process spends in a state can be quite large especially when some of the functions $g_i(x, y, k)$ have a long tail.

To show how our superposition is constructed, let us assume for the moment that only one particular process, say process i, has just experienced a state transition and another process j, $j \neq i$, did not change state at that instant and that it is in state x_j. Let $T(x_j)$ be the time that process j has spent so far in state x_j. Due to the independence of the two processes, the distribution of $T(x_j)$ will be equal to the distribution of the life-time in state x_j. The time $T'(x_j)$ that it takes process j to undergo a state change would have a distribution equal to the residual time distribution at state x_j.

This is illustrated through the example given in figure 1. At the instant marked *observation instant*, process 1 makes a state transition from state 2 to state 3. The distribution of time until processes 2 and 3 experience a state transition, $T'(x_2)$ and $T'(x_3)$ respectively, is approximately equal to the residual life-time distribution in state 2 and state 1 of each process respectively. This is true when we take all possible realization of the above event and assuming that all processes are independent.

We define the superposition state at instants when one or more of the individual processes experience a state transition. The superposition state is described by the tuple $[(x_1, t_1), (x_2, t_2), \cdots, (x_N, t_N)]$ where $x_i \in \{1, 2, \cdots, N_i\}$ is the state of process i observed immediately after a transition occurs, and $t_i \in \{0, 1\}$ indicates whether process i has changed state or not, with $t_i = 1$ iff process i has changed state.

The state space Ξ of the superposition process is given by

$$\Xi = \left\{ [(x_1, t_1), (x_2, t_2), \cdots, (x_N, t_N)] : x_i \in \{1, 2, \cdots, N_i\}, \ t_i \in \{0, 1\}, \ \sum_{i=1}^{N} t_i \neq 0 \right\}$$

Note that $\sum_{i=1}^{N} t_i \neq 0$ because the superposition process is observed immediately after a transition occurs. The number of states in the superposition process is obviously equal to $(\prod_{i=1}^{N} N_i)(2^N - 1)$.

Observation Instant

3

2

1 —————————————— Process 1

2

1 —————————————— Process 2

$T(x_s)$ $T'(x_s)$

2

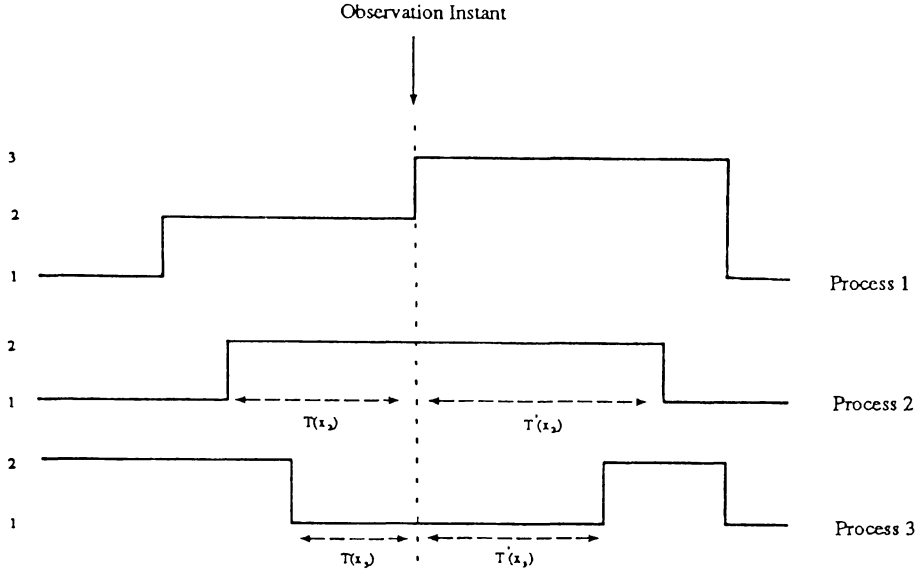1 —————————————— Process 3

$T(x_s)$ $T'(x_s)$

Figure 1: The distribution of residual life-time for a process not experiencing a state change.

In order to fully characterize the superposition process as a MRP, we must obtain the distribution of the time it takes the MRP to move from a state u to state v, where $u, v \in \Xi$ and v is directly accessible from u. Let the probability of going from u to v in k slots be denoted by $q(u, v, k)$. Then $Q = \left[q(u, v, k) \right]$ is the semi-Markov kernel of the superposition process. We now proceed to calculate the functions $q(u, v, k)$.

Let $\hat{g}_i(x, y, k)$ be the residual life-time probability mass function associated with $g_i(x, y, k)$ which is calculated using equation 1. Also, let $G_i(x, k) = \sum_{y=1}^{N_i} \sum_{l=1}^{k} g_i(x, y, l)$, be the cumulative distribution function of sojourn time up to slot k in state x for process i, and $\hat{G}_i(x, k) = \sum_{y=1}^{N_i} \sum_{l=1}^{k} \hat{g}_i(x, y, l)$ be similarly defined for the functions $\hat{g}_i(x, y, k)$.

The function $q(u, v, k)$ depends on the values of $(x_i(u), t_i(u))$ and $(x_i(v), t_i(v))$ for all processes i, where $(x_i(s), t_i(s))$ is the state of process i when the superposition is in state s. We assume that the transition from state u to state v would take place in k slots and then proceed to calculate the probability of this event to occur. Consider an arbitrary process i, based on the values of $t_i(u)$ and $t_i(v)$, the following four distinct cases are possible:

I- $t_i(u) = t_i(v) = 0$. In this case we have $x_i(u) = x_i(v)$, i.e. process i does not change state in either state u or v. The probability of such an event to occur for process i is

6

approximately given by $\phi_i(u, v, k) = 1 - \hat{G}_i(x_i(u), k)$, which is the probability that the residual life-time in state $x_i(u)$ is greater than k.

II- $t_i(u) = 0$ and $t_i(v) = 1$. In this case process i changes state at v but not at u. Since process i has not changed state at u, the probability that this event occurs is $\phi_i(u, v, k) = \hat{g}_i(x_i(u), x_i(v), k)$, which is the probability that the residual life-time in state $x_i(u)$ is equal to k.

III- $t_i(u) = 1$ and $t_i(v) = 0$. In this case process i changes state at u but not at v. The probability that this event occurs is $\phi_i(u, v, k) = 1 - G_i(x_i(u), k)$, which is the probability that the sojourn time in state $x_i(u)$ is longer than k.

IV- $t_i(u) = t_i(v) = 1$. In this case process i changes state at both u and v. The probability that this event occurs is given by $\phi_i(u, v, k) = g_i(x_i(u), x_i(v), k)$, which is the probability that the sojourn time from state $x_i(u)$ to state $x_i(v)$ is equal to k slots.

Thus, due to the independence of the component processes, we have the result

$$q(u, v, k) = \prod_{i=1}^{N} \phi_i(u, v, k) \tag{2}$$

for all $u, v \in \Xi$ and $k \in Z^+$.

We now discuss the computational complexity of the algorithm for constructing the kernel of the superposition process. Let us assume that the maximum length of the probability density function tail in any of the kernels $\{G_i\}_{i=1}^{N}$ is given by L. The storage space needed for the semi-Markov kernel Q to be generated is $O\left(\left(\prod_{i=1}^{N} N_i\right)^2 (2^N - 1)^2 L\right)$. The computational complexity (number of floating point multiplications) is $O\left(\left(\prod_{i=1}^{N} N_i\right)(2^N - 1)^2 L\right)$. This is due to the fact that for an arbitrary state u, the number of states directly accessible from u after one transition is $(2^N - 1)$. Even for a small value of N, say $N = 5$, $N_i = 2 \,\forall i$, and $L = 1000$, the storage needed is $O\left(2^{20} \times 1000 \times L\right)$ which is excessively large for implementation on a conventional computer. By using sparse matrix techniques the storage capacity needed can be reduced to $O\left(\left(\prod_{i=1}^{N} N_i\right)(2^N - 1)^2 L\right)$. Clearly, for a large number of processes, an approximate solution should be sought instead. For example, the MRP may be approximated by a Markov chain or by a renewal process.

The validation of the methodology presented above for characterizing the superposition process is given in section 5.1. It is shown there that the proposed method is quite accurate.

3   The Superposition of Multiple Arbitrary on/off Sources

3.1   The Arbitrary On/Off Source Model

In this section, we introduce the arbitrary on/off traffic source model which is to be used throughout the paper. Consider a traffic source that alternates between active (on) and idle (off) periods. The source always transmits one cell per slot when it is in the on state, see figure 2. For traffic source i, the length of the on and off periods has an arbitrary distribution. For source i, let $f_i^{on}(k)$ and $f_i^{off}(k)$ be the probability that the length of an on and off period is k time slots respectively.

The stochastic process describing the source is in fact an alternating renewal process with two states which can be described by means of a MRP. Let the two states be 0 and 1 corresponding to the off and on state of the source respectively. The associated semi-Markov kernel is given by

$$G_i(k) = \begin{bmatrix} 0 & f_i^{off}(k) \\ f_i^{on}(k) & 0 \end{bmatrix}.$$

We can now directly apply the results of section 2.2 to characterize the superposition process of $N \geq 2$ arbitrary on/off sources. Since $N_i = 2$, $i = 1, 2, \cdots N$, and $x_i \in \{0, 1\}$, the expression for $\phi_i(u, v, k)$ in equation 2 is simplified. For N sources, the superposition process has $2^N(2^N - 1)$ states.

As discussed in section 2.2, this method is limited to small number of input sources, since the time and space complexity grows exponentially with the number of sources. Below, we present an aggregation method for constructing the superposition of arbitrary on/off processes.
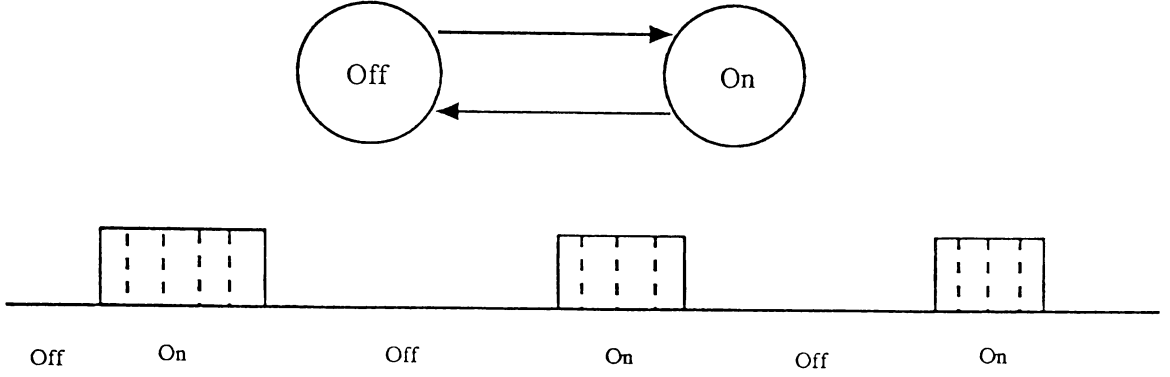
Figure 2: The arbitrary on/off source model.

3.2 On the Aggregation of States in the Superposition of Multiple Arbitrary on/off Sources

Consider a discrete-time MRP with kernel $Q = \left[q(u, v, k) \right]$ and M states. This MRP, referred to as the *original* MRP, is to be approximated by an *aggregate* MRP with kernel $\tilde{Q}$ and $\tilde{M} < M$ states which is equivalent in some sense to Q. Let the states of the original MRP be numbered as $1, 2, \cdots, M$. Let the state space of the original process be divided into the disjoint sets $Z_1, Z_2, \cdots, Z_{\tilde{M}}$, where $Z_i \cap Z_j = \emptyset$, $i \neq j$, and $|Z_i| \geq 1$, $\forall i, j$. All states in set Z_i will be represented by a single state i in the aggregate process. The entries $\tilde{q}(i, j, k)$ of the kernel $\tilde{Q}$ are given by

$$\tilde{q}(i, j, k) = \frac{\sum_{u \in Z_i} \sum_{v \in Z_j} \pi_u q(u, v, k)}{\sum_{u \in Z_i} \pi_u} \tag{3}$$

where $\{\pi_u\}_{u=1}^{M}$ is the invariant probability vector of the underlying Markov chain with probability transition matrix $P = [P(u, v)] = [\sum_k q(u, v, k)]$. It can be shown that the aggregate MRP $\tilde{Q}$ has the same first order characteristics as the original MRP Q. For example the fundamental mean (see [7]) is the same. For the case when the original MRP is the superposition of arbitrary on/off sources, the aggregate process will have the same mean number of arrivals and mean time between instants at which arrivals occur. For this original MRP, an intuitive approach to constructing the aggregate process is to lump all states that have the same number of sources in the on state into a single state. This

suggests an aggregation scheme in which superposition states

$$\left\{ u = [(x_1, t_1), \cdots (x_N, t_N)] : \sum_{i=1}^{N} x_i = c \right\}$$

are lumped to a single state c, where $0 \le c \le N$. In essence we define the sets $Z_c = \left\{ [(x_1, t_1), \cdots (x_N, t_N)] : \sum_{i=1}^{N} x_i = c \right\}$, $c = 0, 1 \cdots N$ and then use equation 3.

In general, the aggregation is difficult to carry out since it requires to first construct the superposition process of N arbitrary on/off processes using the method given in section 2.2. Time and space complexity limits this superposition to about four arbitrary on/off processes. An alternative approach is to construct the aggregate superposition process in a stepwise fashion as follows. First, the aggregate superposition process is initialized to the MRP of the first on/off source. This is followed by an iterative scheme in which the current aggregate superposition process is superimposed with another on/off source using the methodology in section 2.2 at each step. The resulting intermediate process is then aggregated as described above to provide the new aggregate superposition process. The iteration is repeated $N - 1$ times.

The advantage of this algorithm is that the space and time complexity is reduced from an exponential to a polynomial function of the number of superimposed sources. The algorithm is shown in figure 3.

⋄ Let the kernel of the MRP of source i be G_i

⋄ Let $\tilde{Q} = G_1$

⋄ For $i = 2$ to N do

    - Let Q = superposition of $\tilde{Q}$ with G_i
    using the method described in section 2.2

    - Aggregate Q into $\tilde{Q}$ by lumping all states with the same
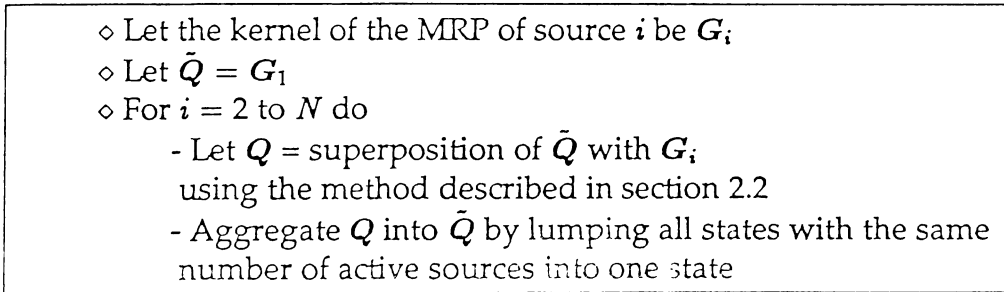   number of active sources into one state

---

Figure 3: The Step-wise aggregation algorithm for constructing the superposition process.

The resulting process has $N + 1$ states. The largest number of states encountered is during the last iteration where we superpose a process an on/off source (2 states) with a process with N states (aggregate superposition of $N - 1$ sources). The resulting intermediate process has $6N$ states according to section 2.2. This is then aggregated to
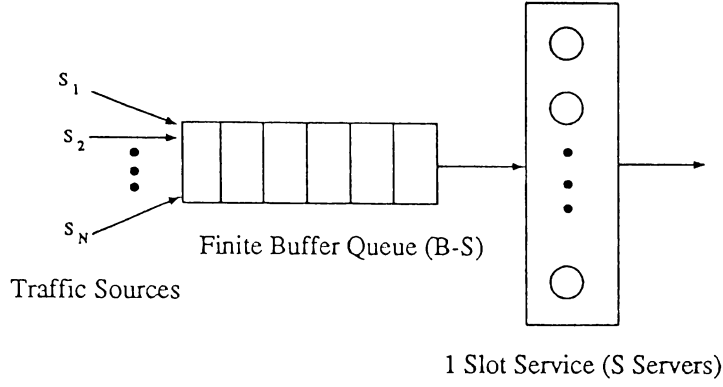
Figure 4: The Statistical Multiplexer Model with S servers and B total waiting capacity.

provide the final approximate process with $N + 1$ states.

The accuracy of this aggregation scheme depends on the characteristics of the input traffic sources and the extent of their homogeneity. The aggregation distorts the statistical properties of the original superposition MRP. Even in the case of completely homogeneous sources, the proposed aggregation scheme may give inaccurate results. However, it is quite accurate when all the input sources are IBP's. Validation results for the aggregation algorithm are given in section 5.3.

4 Analysis of a Statistical Multiplexer with Multiple Arbitrary on/off Input Sources

Consider a FIFO finite buffer multiplexer serving $N \geq 2$ arbitrary on/off sources as depicted in figure 4. Each source is described in terms of the probability density function of its on and off periods. A source emits cells at each time slot when it is in the on state. The multiplexer has $S \geq 1$ servers and can accommodate a total of $B \geq S$ cells at any time instant including those in service. The service time for all cells is constant and is equal to one time slot. The multiplexer can serve S cells every time slot. We assume that $N > S$, otherwise no queue will ever form in the multiplexer and the problem will be trivial to handle.

We seek the steady state probabilities, $\pi(n)$, $0 \leq n \leq B - S$, that there are n cells in the
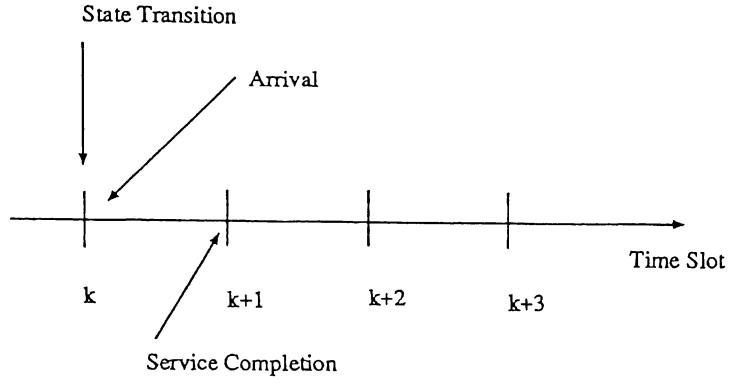
11

Figure 5: Timing of events in the early arrival model.

multiplexer's queue. From this we can obtain other measures of interest such as the mean queue length, the probability of full buffer and the cell loss probability.

Let us first discuss the timing of events in our system. We follow an *early arrival* timing model as defined by Hunt [5]. That is, during an arbitrary time slot, the sequence of events is as follows: a state transition in the superposition may occur, followed immediately by cell arrivals (if any), which is followed by service of waiting cells if there are any, and finally departure of cells that received service. This is shown in figure 5. If an arbitrary cell sees one or more empty servers upon its arrival, it is immediately admitted to one of the available servers without waiting till the next slot. Hence, the multiplexer is effectively of the cut-through type.

The superposition process of the N sources is first characterized as a MRP. Let M be the number of states of the MRP, with states numbered $0, 1, \cdots, M - 1$ $Q = [q(x, y, k)]$ its semi-Markov kernel with M states numbered $0, 1, \cdots, M - 1$. The superposition can be constructed using the method described in section 2.2 or the aggregation method described in section 3.2. Let $a(h)$ be the number of active sources at state $h \in \{0, 1, \cdots, M - 1\}$ of the superposition process.

System State: The system state at any particular slot is described by the pair (n, h) where $0 \leq n \leq B - S$ is the number of cells in the multiplexer buffer at the beginning of the slot (not counting the cells that are to arrive at this slot) and $0 \leq h \leq M - 1$ is the current state of the superposition process. We note here that the process (n, h) at successive time slots does not form a Markov chain. This is due to the non-Markovian nature of the

12

superposition process. However, by observing the system at instants immediately after the superposition process experiences a state transition, the states (n, h) at these instants form an embedded Markov chain since successive states visited by a MRP form a Markov chain.

Solution Method: The probability transition matrix P governing transitions between all possible states (n, h) is first generated. Then the embedded steady state probabilities $\hat{\pi}(n, h)$ are calculated. Finally, the arbitrary point probabilities of observing state (n, h), $\pi(n, h)$, are obtained from $\hat{\pi}(n, h)$. The two fundamental technical difficulties that arise here are generating the matrix P and the calculation of $\pi(n, h)$ from $\hat{\pi}(n, h)$.

Generation of the Probability Transition Matrix: Consider the queue occupancy evolution process at the multiplexer. Assume that the superposition has just made a transition to state h and that the number of cells in the multiplexer immediately before that transition occurs was n_0. During the time interval at which the superposition process is in state h, $a(h)$ cells arrive at the beginning of each slot. At each time slot, if the number of newly arrived cells plus the number already in the system is greater than B, then the excess cells are dropped randomly, i.e. independently of which source they originate from. By the end of a time slot a maximum of S cells in the multiplexer (possibly including those who have just arrived) are served. Assume that the superposition is in state h and that it makes a transition to state h' in k slots. Also, assume that the number of cells in the multiplexer when the superposition process made the transition to state h was n_0. Then, the number of cells in the multiplexer after r slots can be calculated using the following recursive equation:

$$n_r = max\left(0, min(n_{r-1} + a(h), B) - S\right), \ r = 1, 2, \cdots \tag{4}$$

By applying the above equation k times we can find the number of cells in the multiplexer k slots after a transition to state h occurs. By conditioning on the probability that superposition makes a transition from state h to state h' in k steps, we *increment* the probability of going from state (n_0, h) to state (n_k, h') by $q(h, h', k)$. Note that it is possible that a specific state (n', h') can be visited from state (n_0, h) for different values of k. For example if $n_0 = 0$ and $a(h) = 0$, then for all $k \geq 1$, state $(0, h')$ will be the next visited state of the embedded Markov chain. This suggest the algorithm in figure 6 for generating the

13

> ◇ Let $p[(n, h), (n', h')] = 0$ for all states.
> ◇ For all states (n_0, h) do
>   * For all values of k and h'
>   * Find n_k using equation 4
>   * If $q(h, h', k) \neq 0$ then let $p[(n_0, h), (n_k, h')] = p[(n_0, h), (n_k, h')] + q(h, h', k)$

Figure 6: The algorithm for generating the probability transition matrix.

probability transition matrix $P = \left[p[(n, h), (n', h')] \right]$.

Once the probability transition matrix P is generated, we solve for the invariant probability vector $\hat{\pi}(n, h)$ which is the probability of observing the queueing system in state (n, h) given that the superposition process has just undergone a state transition.

Arbitrary-time Probability Calculation: The key to the calculation of the arbitrary-time probability distribution of the queue occupancy is that the system evolution is deterministic given a specific state h of the superposition process, an initial queue occupancy level n_0, and the number of slots k measured from the instant when the superposition process moved to state h.

Let the state of the system at an instant where a transition occurs be (n_0, h). Let us assume that the superposition process makes a transition to state h' after $k \geq 0$ slots with probability $q(h, h', k)$. Then, all states (n_r, h), $1 \leq r \leq k - 1$, where n_r is calculated using equation 4, will be observed with probability one, conditioned on the initial state (n_0, h) and that a transition from state h to state h' occurs in $l > k$ slots. Probabilities $\pi(n, h)$ can then be calculated using the algorithm shown in figure 7. Note the essential normalization step.

Once the arbitrary point probabilities $\pi(n, h)$ have been found, performance measures of the multiplexer like the mean queue length and the cell loss probability can be obtained. The mean queue length can be easily obtained from the probabilities $\pi(n, h)$. The probability of loss $P(Loss)$ is calculated as follows:

$$P(Loss) = \frac{\sum_n \sum_h min(n + a(h) - B, 0)\pi(n, h)}{\sum_n \sum_h a(h)\pi(n, h)} \tag{5}$$

which is equal to the average loss rate divided by the average arrival rate.

14

Figure 7: The algorithm for calculating the arbitrary-time probability

5 Validation Results

In this section, numerical results are presented in order to study the accuracy of the algorithms presented in sections 2.2, 3.2, and 4. The accuracy of these algorithms is assessed by comparing their results with detailed simulation results. First, we present results for the approximation method for characterizing the superposition process of discrete-time MRP's presented in section 2.2. Validation of the statistical multiplexer model follows. The validation of these two algorithms is carried out assuming two MRP's or two on/off sources. The methodology can be used for a small number of processes, say three or four, without running into computational problems. We conjecture that the methodology should remain accurate for than two processes, however, we do not have data to support our claim. Finally, we provide results for testing the accuracy of the stepwise aggregation algorithm for more on/off sources.

5.1 Validation of the Superposition of Discrete-time Markov Renewal Processes

In this section, we report results for testing the accuracy of the approximate characterization of the superposition process of multiple discrete-time MRP described in section 2.2. We consider the superposition of two MRP's with the following structure. Let $g(i, k)$ be the probability that the sojourn time in state i is k slots long, and let $p(i, j)$ be the underlying

probability of going from state i to state j. Then, $g(i,j,k)$ is equal to $g(i,k)p(i,j)$, where $g(i,j,k)$ is the probability that the sojourn time from state i to state j is of length k.

The first MRP has three states, the distribution of the sojourn time in each state is truncated geometric with the parameters shown in table 1. The underlying probability transition matrix is also shown in table 1. The second process has four states, the distribution of the sojourn time in states 0 and 2 is uniform and in states 1 and 3 is modified binomial. The parameters of the distributions and the underlying probability transition matrix are shown in table 2. In both tables 1 and 2, the column marked $Prob(Len = k)$ gives the probability that the sojourn time in a particular state is equal to k.

The approximate superposition process was obtained as described in section 2.2 and it is characterized by a MRP with 36 states. In order to validate the superposition process we conducted a long simulation experiment and observed the statistical properties of the superposition process. In the results shown below, the relative error for a measured quantity x is defined as $|\frac{x(analytic)-x(simulation)}{x(analytic)}| \times 100$, where $x(analytic)$ and $x(simulation)$ is the estimate of x as obtained by analysis and simulation respectively.

The embedded probability of visiting a particular state and the associated relative error is shown in figure 8, the largest error is 0.53%. We also show the mean sojourn time in states of the superposition and the associated relative error in figure 9. The largest relative error is approximately 5.7%. The results in figure 10 show the distribution of sojourn time between two arbitrary source-destination pairs. The first pair is [(1,1),(1,0)] and [(1,0),(2,1)], while the second pair is [(0,0),(0,1)] and [(2,1),(0,0)]. The analytic and simulation results are almost identical.

The approximation model was observed, in general, to be accurate. We have experimented with other types of distributions and different processes and found the accuracy to be reasonable. Also, we observed that the more the states' sojourn time distribution resembles a memoryless distributions, the better the approximation is. This is due to the fact that if all the sojourn time distributions are memoryless the approximate superposition is actually an exact description of the actual superposition process.

State	$Prob(Len = k)$, $k = 1 \cdots L$
0	$(1-\alpha)\alpha^{k-1}/(1-\alpha^L)$, $\alpha = 0.998$, $L = 1000$
1	$(1-\alpha)\alpha^{k-1}/(1-\alpha^L)$, $\alpha = 0.995$, $L = 1000$
2	$(1-\alpha)\alpha^{k-1}/(1-\alpha^L)$, $\alpha = 0.99$, $L = 200$

(a) Sojourn time distribution.

$$\begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.0 & 0.3 & 0.7 \\ 0.6 & 0.3 & 0.1 \end{bmatrix}$$
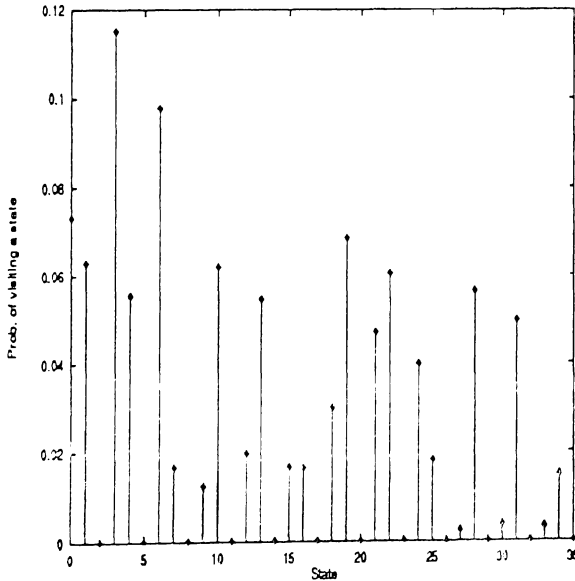
(b) Prob. transition matrix.

Table 1: Parameters for the first MRP.

State	$Prob(Len = k)$, $k = 1 \cdots L$
0	$1/L$, $L = 800$
1	$\binom{L-1}{k-1} \alpha^{k-1}(1-\alpha)^{L-k}$, $L = 100, \alpha = 0.7$
2	$1/L$, $L = 300$
3	$\binom{L-1}{k-1} \alpha^{k-1}(1-\alpha)^{L-k}$, $L = 50, \alpha = 0.3$

(a) Sojourn time distribution.

$$\begin{bmatrix} 0.1 & 0.4 & 0.1 & 0.4 \\ 0.4 & 0.1 & 0.4 & 0.1 \\ 0.3 & 0.2 & 0.3 & 0.2 \\ 0.2 & 0.3 & 0.3 & 0.2 \end{bmatrix}$$

(b) Prob. transition matrix.

Table 2: Parameters for the second MRP.

(a)
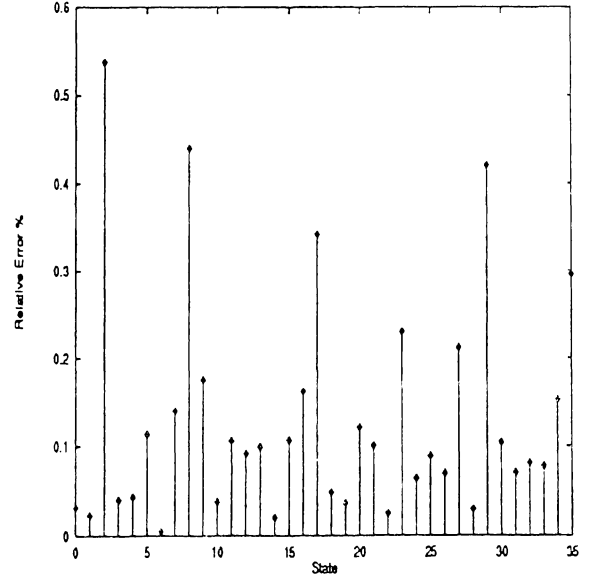
(b)

Figure 8: The embedded probability of visiting one of the states and the relative error.
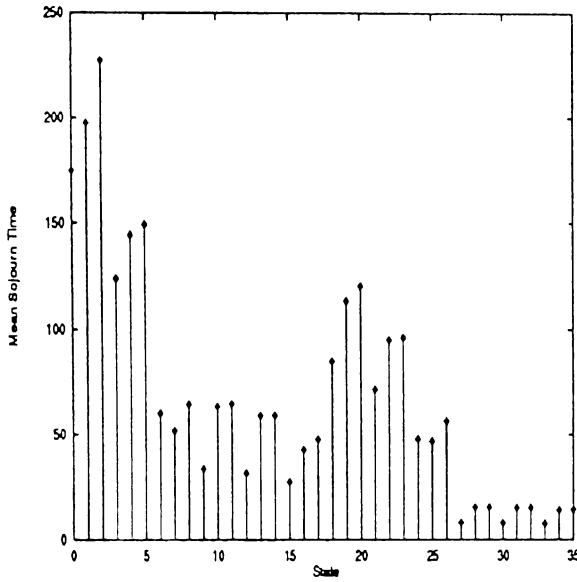
(a)
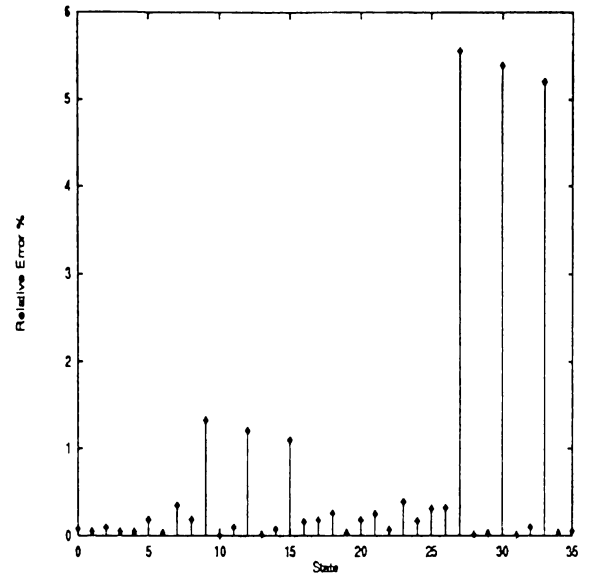
(b)

Figure 9: The mean sojourn time in a state and the relative error.
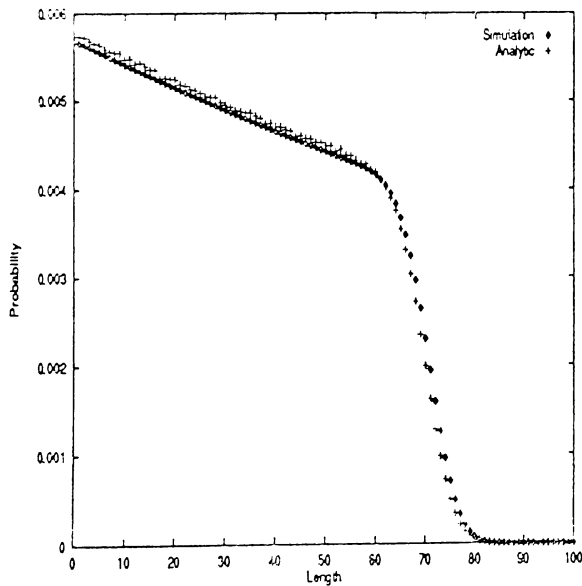
(a) $((1,1),(1,0)) \rightarrow ((1,0),(2,1))$
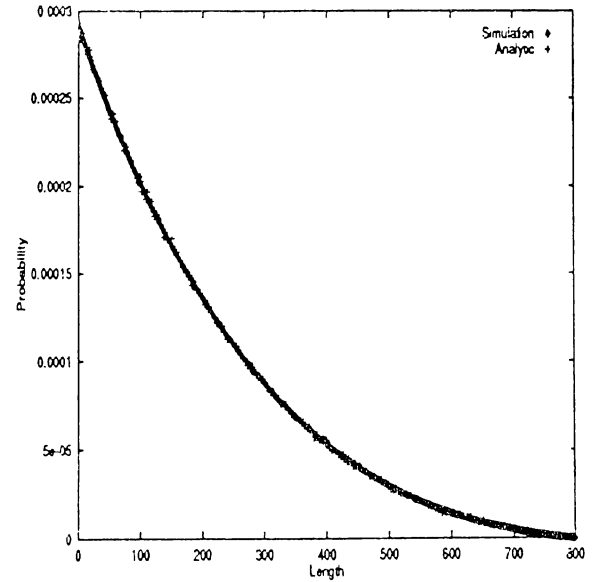
(b) $((0,0),(0,1)) \rightarrow ((2,1),(0,0))$

Figure 10: The distribution of the sojourn time between two arbitrary states.

5.2 Validation of the Statistical Multiplexer Model

In section 4 we presented a stochastic queueing model for the analysis of a finite buffer
FIFO statistical multiplexer whpse input is a MRP which can be the superposition of a
number of independent arbitrary on/off sources. We report results from two experiments
that have been carried out. In the two experiments the buffer size B was taken to be
40 and there was only one server ($S = 1$) in the multiplexer. In the graphs showing the
obtained results, the curves labeled as *detailed, aggregate*, and *simulation* demonstrate the
results obtained by characterizing the superposition process in terms of a 12-state MRP
obtained using the results from section 2.2; by characterizing the superposition process in
terms of a 3-state MRP obtained by the iterative aggregation algorithm; and by simulation
respectively. We note here that confidence intervals for the simulation results are not
shown because they were very narrow. This is also true for the curves shown in the
remaining sections.

Example 1: In the first experiment, we let the multiplexer input be two homogeneous
on/off sources, where the on period is deterministic and the off period is hyper-geometric
with parameters given in table 3. The constant on period length is taken from the set
$\{28, 63, 108, 167, 251\}$ providing a source throughput of $\{0.1, 0.2, 0.3, 0.4, 0.5\}$ respectively.
Figure 11 shows the results for this particular model. We again observe here the proximity
of the results obtained by simulation and detailed superposition. The results obtained by
the aggregate superposition model are less accurate.

|	Mean	CV2
Off period	251	1.5

Table 3: Parameters of off period for results in example 1.

Example 2 - Heterogenous Sources: We tested the effect of source heterogeneity on the
accuracy of the queueing model by studying the case of two *heterogeneous* on/off sources
with hyper-geometric distribution for the on and off periods with the parameters shown
in table 4(a) and table 4(b). For the second source the mean on period $\bar{on}$ is adjusted such
that the source throughput takes values from the set $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$. The
throughput of the first source is fixed to 0.2.

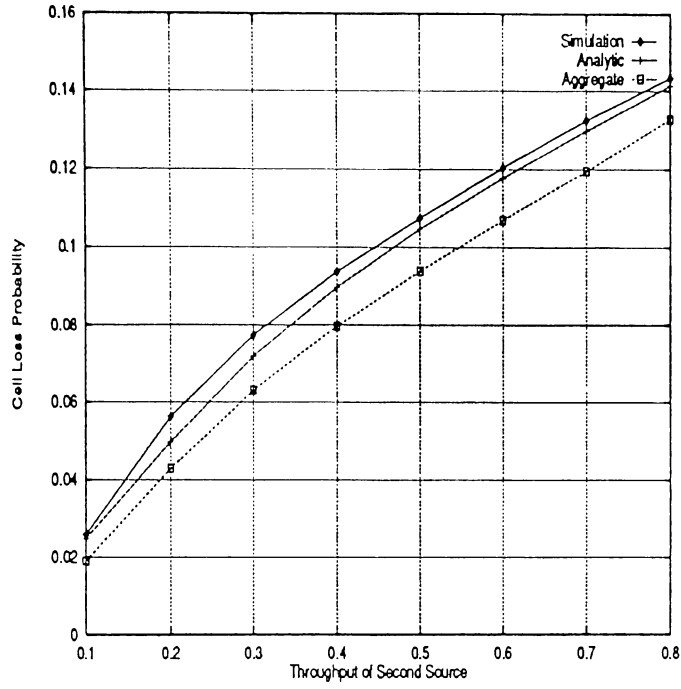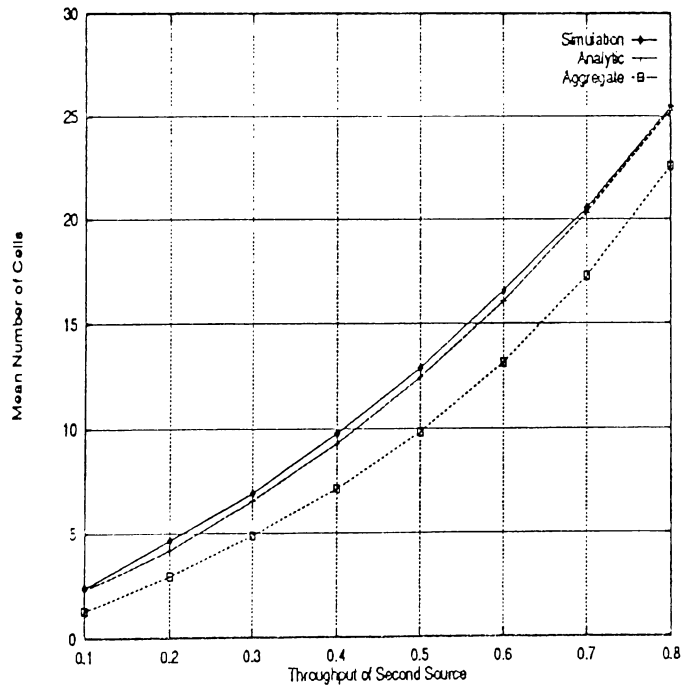In accordance with the results of the previous experiments, we note here that the
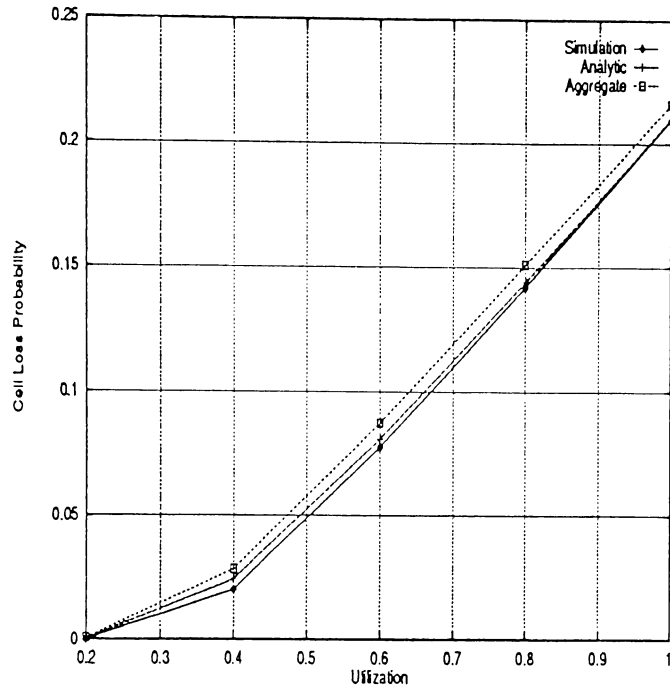
(a)
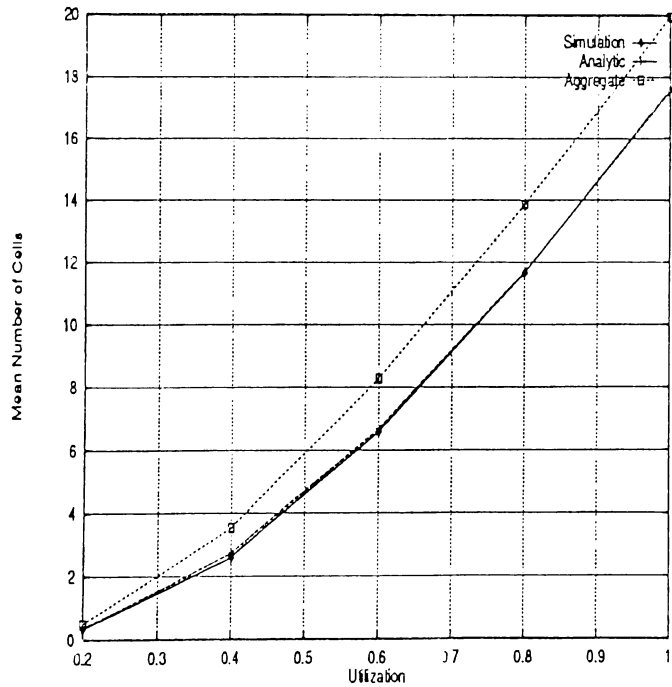
(b)

Figure 11: Validation of statistical multiplexer model. Buffer Size = 40, two input on/off sources with deterministic on and hyper-geometric off periods. (a) Cell loss probability, (b) Mean number of cells.

(a)

(b)

Figure 12: Validation of statistical multiplexer model. Buffer Size = 40, two heterogeneous input on/off sources with hyper-geometric on and off periods. (a) Cell loss probability, (b) Mean number of cells.

detailed model and simulation results are pretty close, while the aggregate superposition model is less accurate. The results are shown in figure 12. In example 1 the aggregate model provided higher values than simulation, whereas in this case, it provides values that are lower than simulation. It is not clear which parameters of the input sources make the aggregation model give higher or lower values than the simulation results. We observed the same behavior in the case of more than two input sources, discussed in section 5.3.

	Mean	CV2
On period	62.5000	5.0
Off period	250	1.5

(a) First source parameters

	Mean	CV2
On period	$\bar{on}$	5.0
Off period	250	1.5

(b) Second source parameters

Table 4: Parameters for example 2.

5.3   Validation of the Iterative Aggregation Algorithm

In section 3.2, we introduced an iterative algorithm based on the notion of aggregation to characterize the superposition process of $N \geq 2$ arbitrary on/off source. We study the accuracy of this algorithm by examining the following three examples.

Example 1: We consider a finite buffer multiplexer with buffer size equal to 40 and one server. All input sources are homogeneous and the on and off periods are hyper-geometric with parameters shown in table 5. The throughput of a single source is equal to 0.0385. The number of input sources was increased from 2 to 20. The simualtion and analytic results for the cell loss probability and the mean number of cells in the multiplexer queue are shown in figure 13.

	Mean	CV2
On period	10.0	5.0
Off period	250.0	1.5

Table 5: Parameters of on and off periods.

These results indicate that the aggregation method does not have a good accuracy. Note that the larger the number of sources, the larger the relative error in the analytic

results. For this particular example, the iterative aggregation method seems to smooth out the original superposition process by making it less bursty and less correlated. This effect is magnified as the number of sources increases, because of the errors introduced at each stage of the aggregation method. The aggregate superposition, however, still captures the first order characteristics of the real superposition process and hence the multiplexer witnesses a longer queue buildup and a larger cell loss probability as the number of sources increases.

Example 2: We consider the same multiplexer parameters and we let the on/off source have a deterministic on period with length equal to 50 and a geometric off period with mean equal to 1101.484. The throughput of a single source is equal to 0.0434. The number of input sources was increased from 2 to 20. The simualtion and analytic results are shown in figure 14. We observe a similar behavior as in the first example. However, in this case the analytic results are closer to the simulation data than in the first example and they provide an upper bound of the cell loss probability and the mean number of cells.

Example 3: We consider a finite buffer multiplexer with buffer size equal to 60 and one server. All input sources are homogeneous and the on and off periods are geometric with means 11.77 and 223.23 respectively. The throughput of a single source is equal to 0.05. The number of input sources was increased from 2 to 20. The results are shown in figure 15. We note that the aggregation method is accurate in the case of Interrupted Bernoulli Processes. We can then conclude that the inaccuracy in the previous cases is not due to the successive iterations but mainly due to the aggregation process which distorts the statistical characteristics of the original superposition process. In the case of heterogenous IBP's, the accuracy of the aggregation is acceptable if the ratio of maximum to minimum squared coefficient of variation of the inter-arrival time is small, as demonstrated in [3].

6   Study of the Effect of Traffic Parameters on Queueing Performance

In this section, we conduct a study of the effect of various traffic parameters on the performance of the statistical multiplexer.
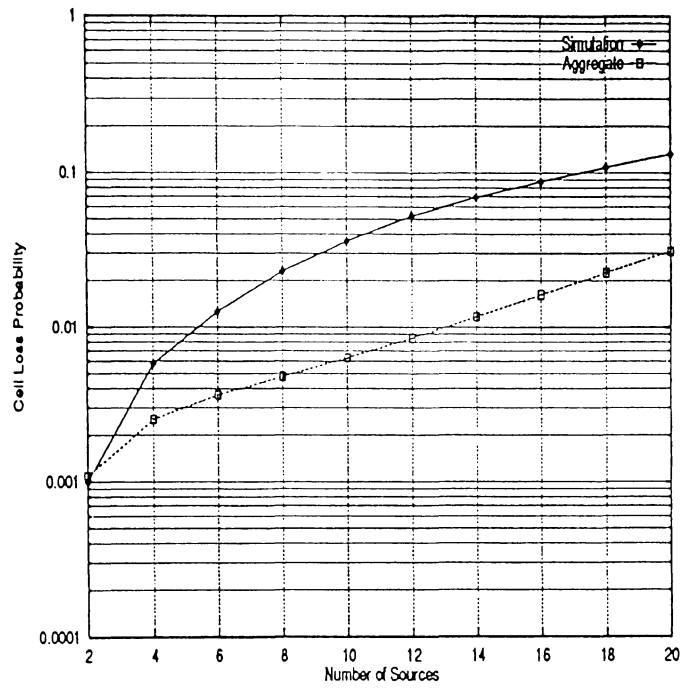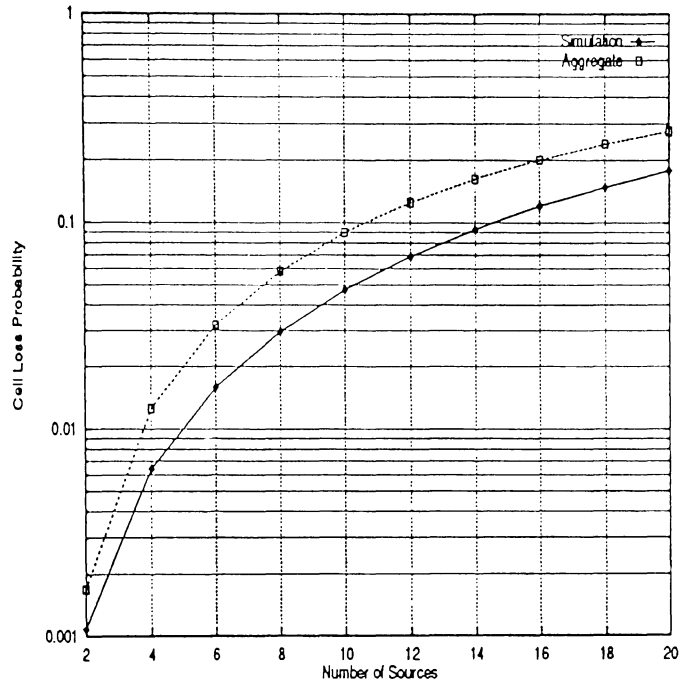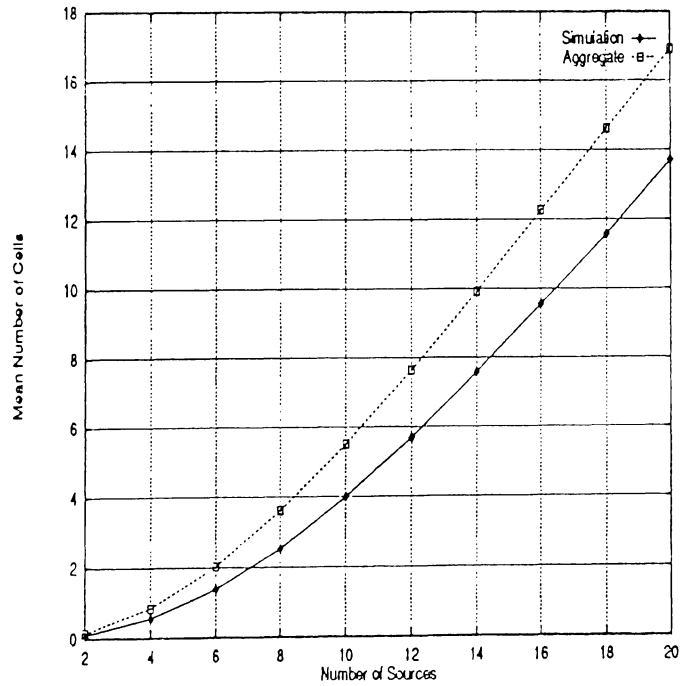
(a)
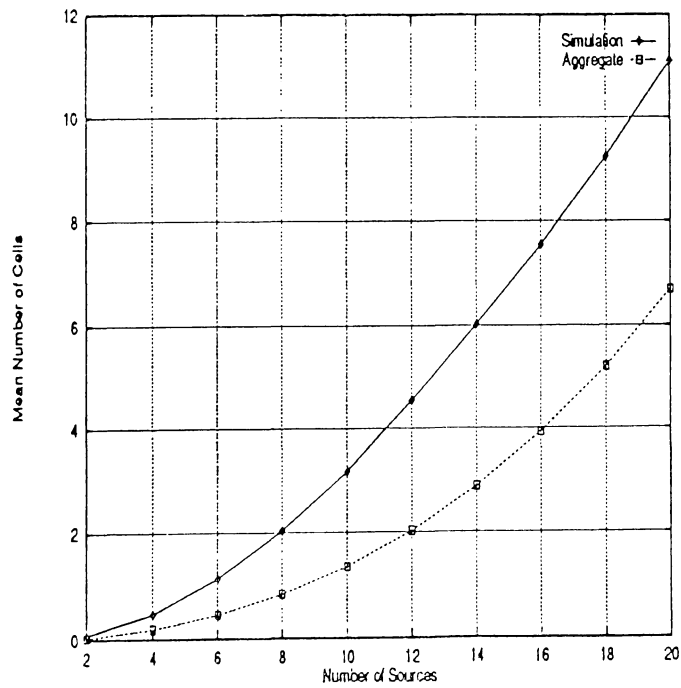
(b)

Figure 13: Validation of the iterative method. Buffer Size = 40. All sources are identical and have a hyper-geometric distribution for the on and off periods. (a) Cell loss probability, (b) Mean number of cells.
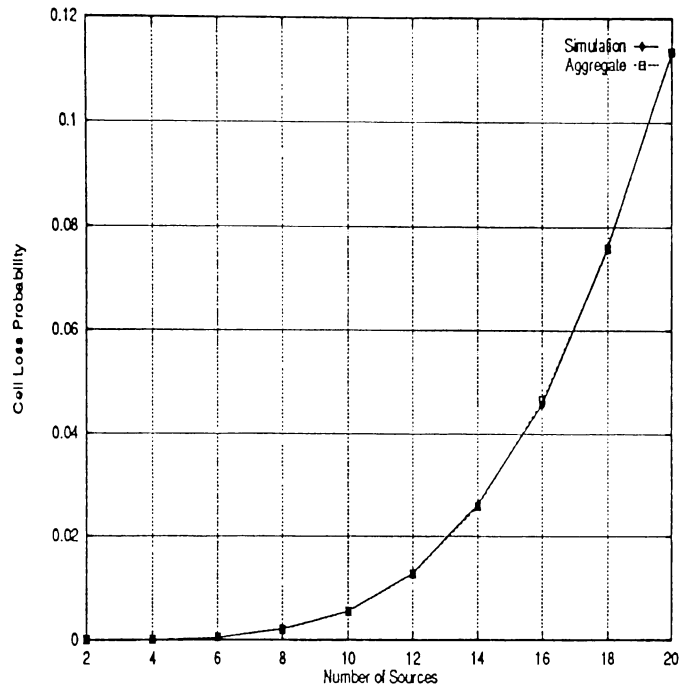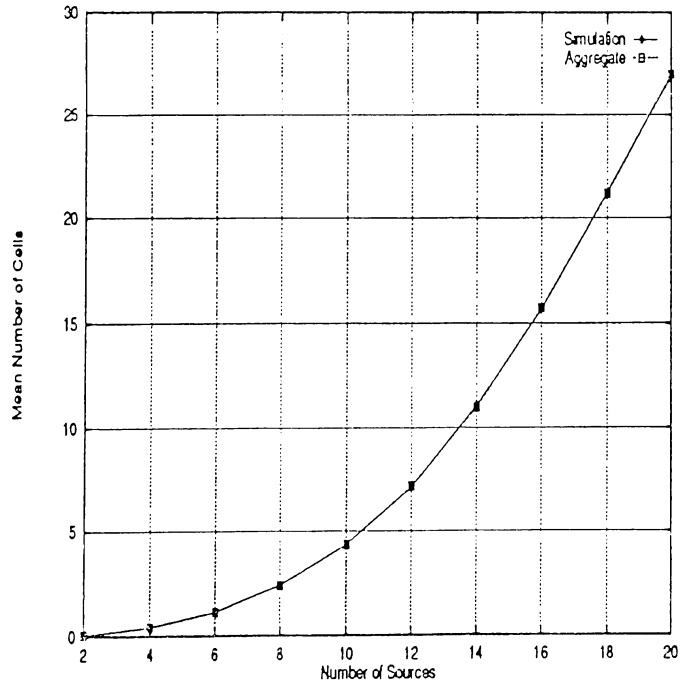
(a)

(b)

Figure 14: Validation of the iterative method. Buffer Size = 40. All sources are identical with a deterministic on period and a hyper-geometric off period. (a) Cell loss probability, (b) Mean number of cells.

(a)

(b)

Figure 15: Validation of the iterative method. Buffer Size = 60. All sources are identical with a geometric on and off periods. (a) Cell loss probability, (b) Mean number of cells.

6.1 Effect of the Distribution of the On and Off Periods on the Multiplexer's Performance

In this section, we study the effect of the distribution of the on and off periods on the multiplexer's behavior. In the literature, it is common to approximate the distribution of sojourn times in a state by a geometric or a hyper-geometric distribution. The geometric distribution characterization requires only the first moment, while the hyper-geometric distribution characterization requires the first two moments. The question that usually arises is whether these approximations are accurate. We consider an extreme case where the original input source is correlated and then study how the multiplexer's performance would change if the source is replaced by an IBP (sojorun times are geometric) or by a source with hyper-geometric on and off periods. Note that in these two models the inter-arrival times are uncorrelated. We consider a single server multiplexer with two homogeneous input sources and a buffer size taken from the set $\{10, 20, 30, 40, 50, 60\}$.

The distribution of the on and off periods of the original input source is a mixture of two deterministic distributions. The inter-arrival time lag-1 correlation is equal to 0.048. The parameters of the pdf of the on and off periods are shown in table 6. The throughput of a single source is equal to 0.296, the mean on and mean off periods are 50 and 119 respectively, and the CV_{on}^2 and the CV_{off}^2 are 2.332 and 13.150 respectively. Given these values we can approximate the original source by an IBP source and a source with a hyper-geometric on and hyper-geometric off periods.

On period		Off period	
Length	Prob.	Length	Prob.
1	0.708333	20	0.95
169	0.291667	2000	0.05

Table 6: Parameters of the pdf of the on and off periods.

In figure 16 we plot the mean number of cells and cell loss probability for the original source, the single-moment approximation by an IBP source, and the two-moment approximation by a source with the hyper-geomtric on and off periods. The IBP model underestimates the cell loss probability and overestimates the mean number of cells (except when the buffer size increases above 52). This suggests that the IBP is not a faithful model for the original source of table 6.

Sohraby [8] introduced a model for handling general on/off sources. The model gives an approximate upper bound for the cell loss probability as a function of the first two moments of the on and off periods assuming multiplexer with an infinite buffer size. This suggests that only the first two moments of the one an off periods affect the performance. However, the results in figure 16, demonstrate the inaccuracy of the two-moment approximation when the buffer size is finite. The approximate source model with two-moments matching provides an underestimation of the cell loss probability and mean number of cells. This shows that the two-moment approximation may not be accurate in all cases.

6.2 Effect of the Interrival-time Correlation of Traffic Sources on the Multiplexer's Performance

For an arbitrary on/off source, the lag-1 correlation coefficient of the inter-arrival time is given by Galmés [4]:

$$\phi_1 = \frac{f_{on}(1) - \frac{1}{\bar{on}}}{1 + CV_{off}^2 - \frac{1}{\bar{on}}}$$

where $f_{on}(1)$ is the probability that the on period is of length 1, $\bar{on}$ is the mean on period, and CV_{off}^2 is the squared coefficient of variation of the off period length. We have been able to identify some distributions for which the value of ϕ_1 is non-negligible. The key to obtaining such distributions is to concentrate a large portion of the probability mass at length 1 of the on period distribution, i.e. make $f_{on}(1)$ as large as possible while satisfying some of the other source characteristics (e.g. given values of $\bar{on}$ and/or CV_{on}^2.) One such distribution is the *mixture of two deterministic distributions* which is a distribution that can be of length L_1 or L_2 with probabilities p and $1 - p$ respectively. We fix one of the deterministic lengths to be equal to 1 and let the other be of a variable length L. Given a particular value of $\bar{on}$ and ϕ_1 and the off period distribution, we can find values for p and L which would satisfy the given values of $\bar{on}$ and ϕ_1 using a simple enumerative algorithm.

To study the effect of the inter-arrival time correlation on the multiplexer behavior, we consider the case of two input homogeneous sources where the off period of a source has a geometric distribution with mean 92.7787 and the mean length of the on period
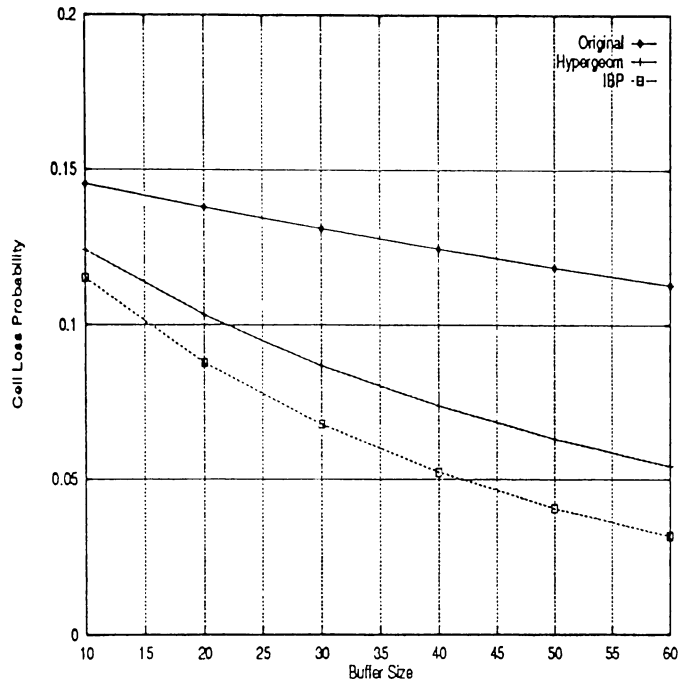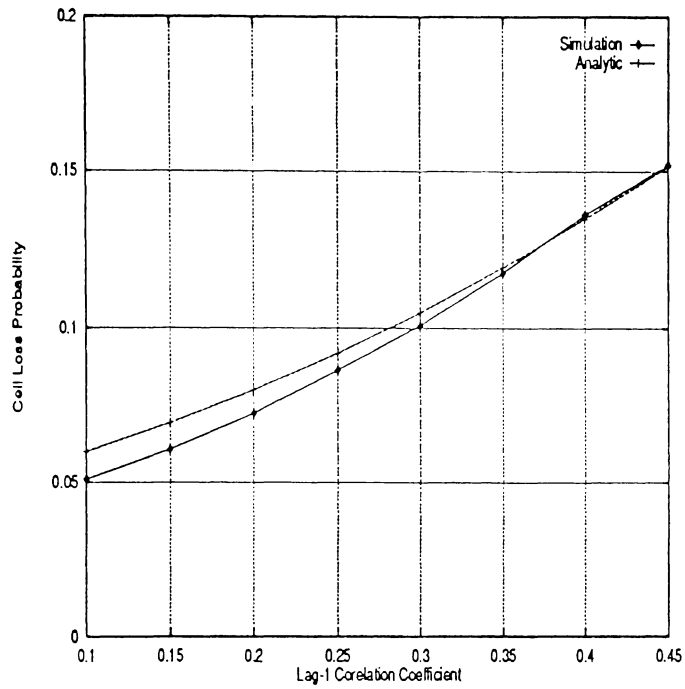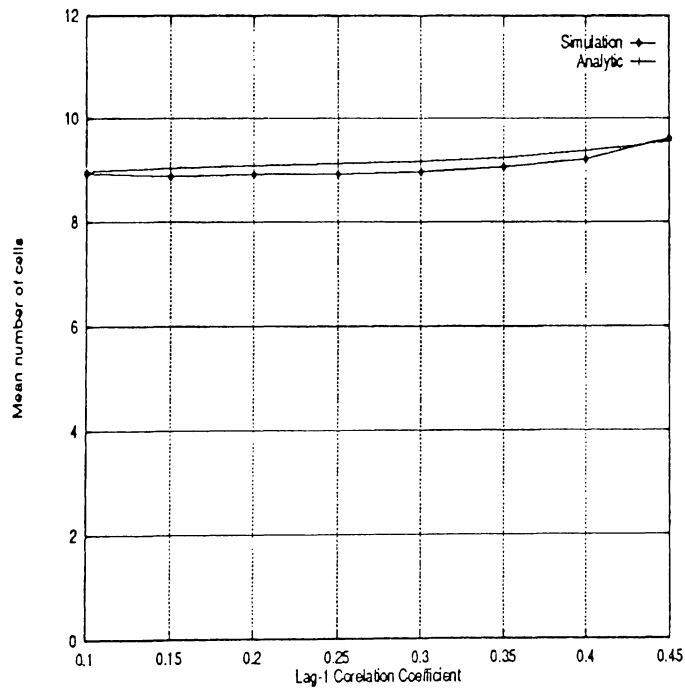
(a)
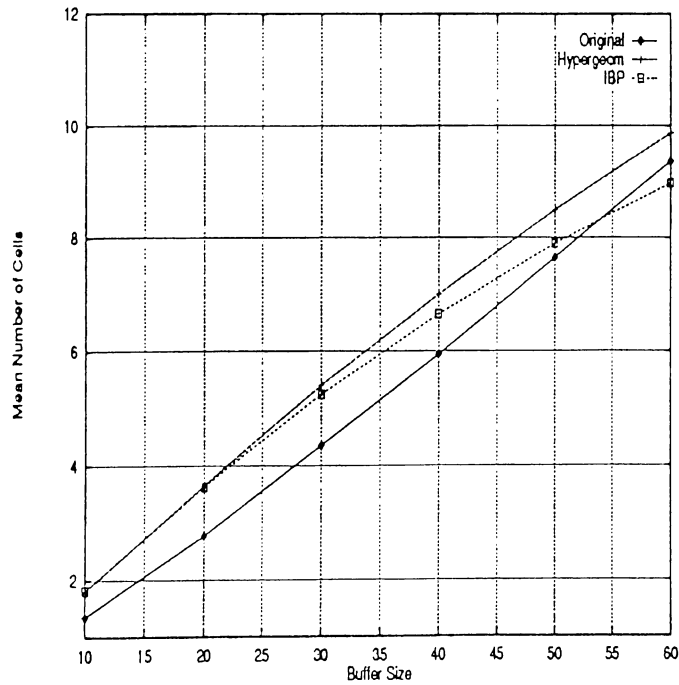
(b)

Figure 16: Effect of the distribution of the on and off periods on the multiplexer's performance. (a) Cell loss probability, (b) Mean number of cells.

Figure 17: Effect of correlation on the multiplexer's performance. Buffer Size = 40, two input on/off sources with hyper-geometric on and off periods. (a) Cell loss probability, (b) Mean number of cells.

is fixed at 50, making the source's throughput equal to 0.35. Using the mixture of two deterministic distributions for the on period, we vary ϕ_1 so that it takes values from the set $\{0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.45\}$. The value of p and L satisfying the given parameters is then calculated. The cell loss probability and the mean number of cells in the multiplexer queue are shown in figure 17.

We note that by increasing the lag-1 correlation, the cell loss probability and the mean number of cells increase. As it can be seen from figure 17, the cell loss probability increases more sharply than the mean number of cells with the increase of the correlation coefficient. The mean number of cells is almost constant and increases very slowly with the increase of the correlation coefficient.
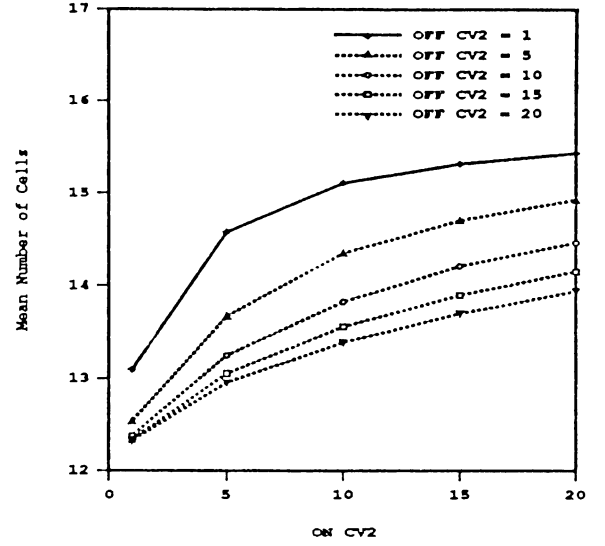
6.3 Effect of the Squared Coefficient of Variation of the on and off periods on the Multiplexer's Performance

Using the versatile hyper-geometric distribution we studied the effect of the squared coefficient of variation of the on and off periods, respectively CV_{on}^2 and CV_{off}^2, on the multiplexer's performance. By specifying the mean and squared coefficient of variation of the period length, it is possible to fit a hyper-geometric distribution given some conditions are met by the specified mean and coefficient of variation (see [2] for more details).

We considered two input sources each with a throughput fixed at 0.4. The mean on and off periods were fixed at 100 and 150 respectively. CV_{on}^2 and CV_{off}^2 take values from the set $\{1.0, 5.0, 10.0, 15.0, 20.0\}$. The results obtained from simulation and detailed superposition are shown in figure 18. Note that by increasing the CV_{on}^2, while CV_{off}^2 is kept constant, both the cell loss probability and mean number of cells increase (see parts (a) and (b) of figure 18). Also, note that the rate of increase of the mean number of cells and cell loss probability when $CV_{on}^2 \in [1, 5]$ is larger than for the rest of the values. Moreover, for larger values of CV_{off}^2, the rate of increase of cell loss probability and mean number of cells as a function of CV_{on}^2 is relatively slower than for smaller values of CV_{off}^2. A surprising result is observed when varying CV_{off}^2 while CV_{on}^2 is kept constant (see parts (c) and (d) of figure 18). As CV_{off}^2 increases, the cell loss probability *increases* while the mean number of cells *decreases*.
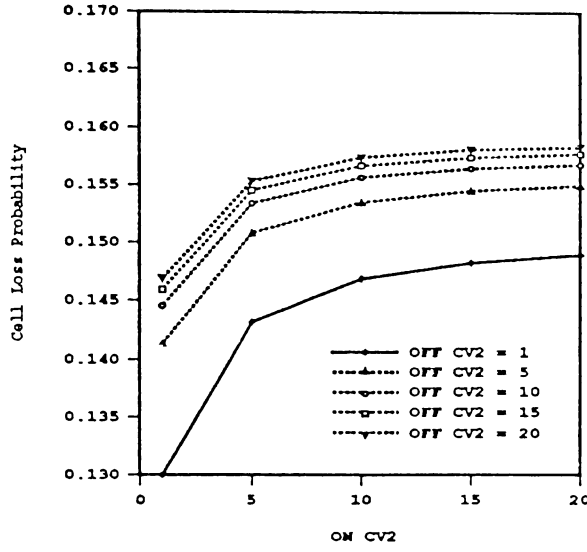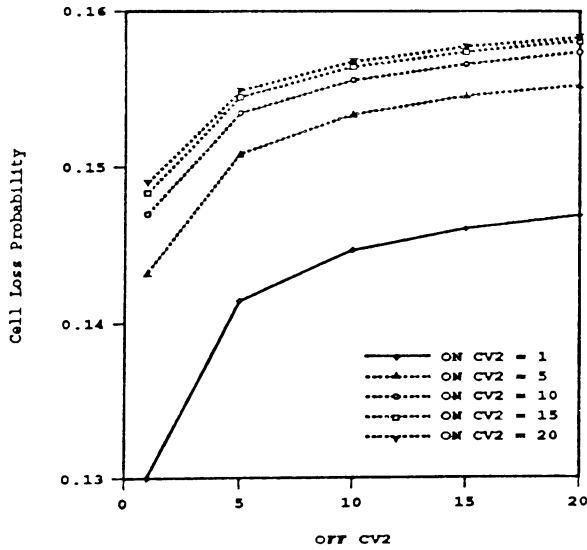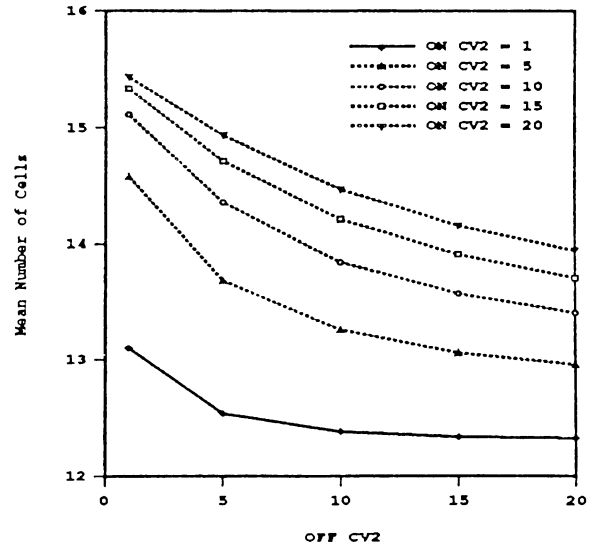
Figure 18: Effect of the squared coefficient of variation of the on and off periods on multiplexer's performance. Buffer Size = 40, two input on/off sources with hyper-geometric on and off periods. (a) Cell loss probability vs. CV_{on}^2. (b) Mean number of cells vs. CV_{on}^2. (c) Cell loss probability vs. CV_{off}^2. (d) Mean number of cells vs. CV_{off}^2.

7 Conclusions

In this paper we presented a new approximation method for characterizing the superposition of multiple independent Markov Renewal Processes. The presentation here focused on discrete-time processes, but the methodology is readily applicable to continuous-time processes with little modification. The model developed for characterizing the superposition can be directly applied to the area of tele-traffic engineering and statistical multiplexing. One special case of this model is the arbitrary on/off source that was used in this paper. We have presented a queueing model for the analysis of a statistical multiplexer whose input is a MRP representing the superposition of multiple traffic sources.

The advantage of our methodology is that it provides a uniform framework in which a variety of models of traffic sources can be handled. The basic limitation is the huge state space and the computational complexity of the algorithms. These disadvantages can be overcome if an elegant state aggregation scheme is found such that the resulting aggregate process has a number of states which is preferably a linear function of the number of sources while still preserving the characteristics of the real superposition process. We introduced an aggregation scheme for reducing the dimensionality of the superposition process. However, during the aggregation, the statistical properties of the original process may be distorted and may subsequently lead to inaccurate results for the multiplexer's performance metrics.

The work presented here introduces many challenging issues. An important problem that is yet to be considered is the characterization of the departure process from a multiplexer with multiple arbitrary on/off input sources, or in general, with an MRP input. Characterizing the departure process of one of the input sources is even more challenging. The methodology presented here introduced a new versatile modeling device for arrival processes which can be called the Batch Semi-Markov Arrival Process (B-SMAP). This is a semi-Markov process with as many states as necessiated by the physical source being modeled, and in which the sojourn times in states are arbitrarily distributed. This process is analogous to the well-known BMAP. However, it is more flexibile than a BMAP but more complex to analyze.

References

[1] W. P. Cherry and R. L. Disney. The Superposition of Two Independent Markov Renewal Processes. *Zastosowania Matematyki*, 17:567–602, 1983.

[2] T. E. Eliazov, V. Ramaswami, W. Willinger, and G. Latouche. Performance of an ATM Switch: Simulation Study. In *Proceedings of the IEEE Infocom*, pages 644–659, 1990.

[3] K. M. Elsayed and H. G. Perros. A Computationally Efficient Algorithm for Characterizing the Superposition of Multiple Heterogeneous Interrupted Bernoulli Processes. Submitted for publication, 1994.

[4] S. Galmés. Analysis of On-Off Processes With Independent Arbitrary Distributions. Unpublished technical report, North Carolina State University, 1992.

[5] J. Hunter. *Mathematical Techniques of Applied Probability. Volume 2: Discrete Time Models: Techniques and Applications*, chapter 9. Academic Press, 1983.

[6] V. S. Korolyuk. Superposition of Markov Renewal Processes. *Cybernetics*, 17:556–560, 1981.

[7] M. F. Neuts. *Matrix-Geometric Solution in Stochastic Models: An Algorithmic Approach*. John Hopkins Univ. Press, 1981.

[8] K. Sohraby. On the Theory of General On-Off Sources With Applications in High-Speed Networks. In *Proceedings of the IEEE Infocom*, pages 401–410, 1993.

[9] H. M. Taylor and S. Karlin. *An Introduction to Stochastic Modeling*, chapter 7. Academic Press, 1984.