Performance Analysis of a Multi-Buffered Banyan ATM Switch Under Bursty Traffic

Todd D. Morris

Harry G. Perros

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

TR-91/1
January 1991
(Revised August 1991)

Performance Analysis of a Multi-Buffered Banyan ATM Switch Under Bursty Traffic

Todd D. Morris[1,2]

Department of Electrical and Computer Engineering, and
Center for Communications and Signal Processing
North Carolina State University
Raleigh, N.C. 27695-7911.

Harry G. Perros[2]

Department of Computer Science, and
Center for Communications and Signal Processing
North Carolina State University
Raleigh, N.C. 27695-8206.

January 1991

Abstract

We present an analytical model of a buffered Banyan ATM switch which allows complex switching elements, bursty traffic, non-uniform destination distributions and permits the analysis of large scale switches. The ATM switch is analyzed by decomposing it into individual switching elements. Each switching element is then analyzed numerically in isolation assuming that its arrival and service processes are known. The parameters of the arrival and service processes of the switching elements are obtained using an iterative scheme. The results obtained are approximate and validation tests have shown that they have good accuracy. Using this model, the cell loss, throughput, and the mean time to traverse the switch were obtained for different traffic parameters, and buffer sizes within a switching element.

[1] Supported by the Natural Sciences and Engineering Research Council of Canada .

[2] Supported in part by DARPA under grant no. DAEA 18-90-c-0039.

1. Introduction

The Asynchronous Transfer Mode (ATM) is a suggested solution for packet switching in broadband ISDN. ATM provides the capabilities of switching bursty traffic such as voice, video and bulk files (see [1]). Multistage interconnection networks have been proposed as a possible switch fabric for ATM. The self-routing capabilities of such networks coupled with their low complexity are desirable features for ATM.

One of the multistage interconnection network that has been considered for ATM switching is the Banyan. This type of switch is typically composed of logn stages of 2x2 switching elements, where n is the number of switch inputs. These switching elements may or may not have buffering. In the unbuffered case, cell loss occurs within a switching element when both inputs compete for the same output port. In buffered switching elements, generally, there is no cell loss. However, cell loss may occur at the input ports of the Banyan switch. The arrangement of the buffers within a switching element greatly affects its operation and performance. Traditionally, the buffers have been lumped together in a FIFO manner at either the input or the output ports of the switching element. However, neither placement is ideal. Input port buffering suffers from head-of-line blocking. That is, the cell at the head of the queue may block the other cells in the queue despite the fact that they may be destined for a free port. It has also been shown, that in this buffering scheme, increasing the amount of buffering only improves performance minimally (see [2]). Output port buffering alleviates this problem since the arriving cells are first switched to their appropriate output ports and then they are placed in the FIFO buffers. Unfortunately, there should be two free buffer spaces in each output buffer before the switching elements in the previous switch stage are able to transmit (this will prevent cell loss). Other more complex buffer architectures have been proposed, but their analysis becomes far more complex (see [3], [4], and [5]).

Due to the recent interest in multistage switch fabrics for broadband ISDN, many performance models for such switches have been proposed. Initially, these models were restricted to unbuffered Banyan switches with identical Bernoulli input sources and uniform destination distribution (see [6]). These models were further modified for single-buffered switch architectures (see [7]). Multi-buffered analysis followed (see [8], [9], and [10]). The multi-buffer switch architectures that were analyzed, however, involved simple FIFO buffering at either the input or the output ports of a switching element but not at both.

In order to accurately model the performance of these switches, it is necessary to assume bursty arrivals and a non-uniform destination distribution. Several models have been reported in the literature which include a non-uniform destination distribution as well as more complex switching element architectures (see [10], [11], and [12]). However, these models were developed assuming identical Bernoulli input sources. Bursty traffic has been used in performance models of buffered Banyan switches, but for simpler types of architectures. Also, models of complex buffered Banyan switches typically have been limited to buffer sizes equal to a couple of cells. Due to the complex nature of the buffered Banyan switch, these previous models incur significant error when the offered loads exceed 0.6 (see [8], [9], [10], and [11]). In some cases, the validity of the models is not known seeing that simulation comparisons were not reported.

In this paper, we present an analytic model of a buffered Banyan switch which allows complex switching elements, bursty traffic, non-uniform destination distributions, and permits the analysis of large scale switches. The results obtained from this model are approximate, and validation tests have shown that it has very good accuracy. In the following section, we describe the structure of the buffered Banyan network under study. The analytic model is presented in section 3, and numerical results are given in section 4. Finally, the conclusions are given in section 5.

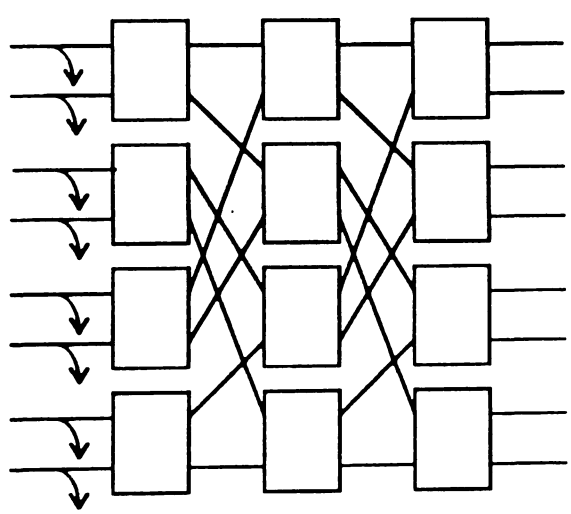

Figure 1: 8x8 Banyan Network

2. The Buffered Banyan Switch Under Study

An 8x8 Banyan switch is shown in figure 1. The buffered switching elements are 2x2 crossbars as shown in figure 2. Each input port of the switching element has a buffer of length one, referred to as the *input buffer*, and each output port consists of a FIFO buffer of length N referred to as the *output buffer*. We distinguish between the two input buffers by referring to them as the *upper* and the *lower* input buffer. Similarly, we shall refer to the two output buffers as the *upper* and the *lower* output buffer. Let r_{uu}, r_{ul}, r_{lu}, and r_{ll} be the branching probability from the upper input buffer to the upper output buffer, from the upper input buffer to the lower output buffer, from the lower input buffer to the upper output buffer, and from the lower input buffer to the lower output buffer respectively.

This buffering scheme has been suggested and analyzed by simulation in [3], [4], and [5]. This scheme avoids the problems associated with exclusive FIFO buffering at either the input or the output ports, such as head-of-line blocking and the need for two free buffer spaces, without affecting the cost of the switching element. Note that other techniques such as cell bypass (see [2]) can be used to alleviate these problems.

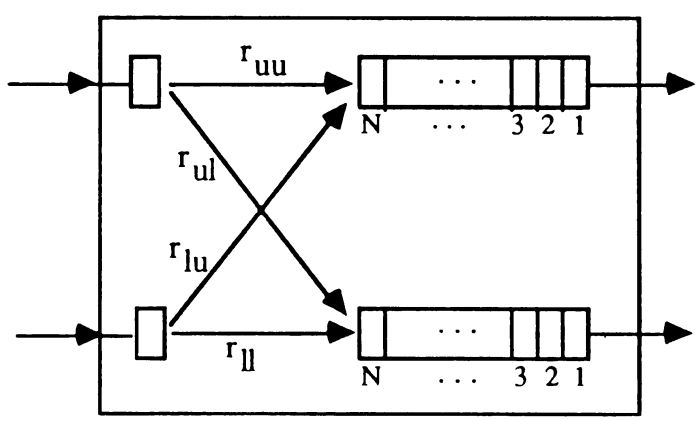

Figure 2: 2x2 Buffered Switching Element

Buffered Banyan switches provide synchronous switching with a back-pressure mechanism which prevents cell loss. Let us define the clock cycle which propagates a cell from one stage of the switch to the next as the *switch cycle*. Cells arrive at the switch in an

asynchronous manner but they do not enter the switch until its next switch cycle. The basic operation of the Banyan network is as follows:

1. Upon arrival to the switch, a cell waits until the next switch cycle. At that time, if the corresponding input buffer is empty, the cell is forwarded to this buffer. If the buffer is full then the cell is lost.
2. Once the cell is buffered at the input of the switching element, the leading bit specifies the desired output port. Assuming that there is sufficient buffer space at the output port, the cell is propagated forward while stripping the leading bit. Note that if both input buffers contain cells directed to the same output, they may both proceed in a random order if there are two free spaces in the output buffer. If one or no buffer space exists at the output buffer, then only the appropriate number (1 or 0) input cells can be forwarded.
3. Once a cell reaches the head of the output buffer, it will be forwarded to the next stage if the corresponding input buffer is free. If the input buffer is not free, the cell will wait until the next switch cycle. If at that time the input buffer is free, the cell will be forwarded to the next stage. Else, it will wait for another switch cycle. This will repeat until the cell enters its destination input buffer. (This mechanism prevents cell loss within the Banyan network.)

Given that a switching element i is empty and that the input buffer of its corresponding switching element j in the next stage is free, a cell will be propagated from i to j in one switch cycle. Also, all buffers function so that a full buffer will accept another cell if the head of that buffer is departing at the same time.

The arrival process to each input port of the Banyan switch is assumed to be bursty and it is modelled by an Interrupted Bernoulli Process (IBP). That is, the incoming link into an input port of the switch is slotted. Each slot is assumed to be equal to the switch cycle. A slot is long enough to contain one cell. An arriving slot may or may not contain a cell. In an IBP, we have a geometrically distributed period during which no arrivals occur (idle state), followed by a geometrically distributed period during which arrivals occur in a Bernoulli fashion (active state). Given that the process is in the active state respectively idle state at slot t, it will remain in the same state in the next slot t+1 with probability p respectively q. During the active state, a slot contains a cell with probability α. The quantity α is also known as the peak bandwidth, i.e. the rate of arrivals during the active

period. It can be shown (see for instance [13]) that the average arrival rate, i.e. the probability that any slot contains a cell, is

$$\rho = \frac{\alpha(1-q)}{2 - p - q},$$

and the squared coefficient of variation, C^2, of the time between successive arrivals is

$$C^2 = 1 + \alpha\left(\frac{(1-p)(p+q)}{(2-p-q)^2} \quad 1\right).$$

The average arrival rate is otherwise known as the average bandwidth.

3. The Model

Conceptually, we view the entire switch as a queueing network where each node represents a switching element. External arrivals occur to the input buffers of the switching elements in stage 1. Subsequently, the departure process of the output buffer of a switching element becomes the arrival process to the input buffer of a switching element in the next stage. Eventually, cells depart when they reach the output buffer of a switching element in the last stage. This type of queueing network is conceptually similar to queueing networks with finite capacity queues. In such queueing networks, due to the finite capacity of the queues, the flow of customers through one queue may be momentarily stopped if a destination queue becomes full. In particular, a customer may be forced to wait in its queue until a space becomes available at the destination queue. In this case, we say that the customer becomes *blocked*. The set of rules that dictate when a customer becomes blocked and how it becomes unblocked is known as the *blocking mechanism* (see [14]). As it was mentioned in the previous section, once a cell reaches the head of the output buffer, it is forwarded to a switching element in the next stage if its corresponding input buffer is free. If the input buffer is not free, the cell is *blocked* and it cannot proceed. The cell will wait until the next switch cycle, and if at that time the input buffer is free, it will be forwarded to the next stage. Else, it will wait for another switch cycle, and so on. Blocking of cells also occurs within a switching element. A cell in an input buffer cannot enter its destination output buffer if the latter is full. In this case, we say that the cell is blocked. The cell is forced to wait in the input buffer for another switch cycle before it attempts to enter the

output buffer. Blocking will also occur if both input buffers contain a cell destined to the same output buffer and the output buffer is either full or it has one free space. In the terminology of queueing networks with finite capacity queues, this type of blocking is known as *blocking-before-service.*

We analyze the switch under study by decomposing it into the individual switching elements. Each switching element is then analyzed independently of the other switching elements. In order to accurately analyze a switching element in isolation, we need to characterize the arrival process to each input buffer of a switching element. This is done using the departure process from the output buffer of the corresponding switching element in the previous stage. Since it takes one switch cycle to forward a cell from an output buffer to the next switching element, we can think of the output buffer as being served by a server which has a constant service equal to one switch cycle. Due to blocking, we distinguish between the *actual* and the *attempted* departure process. The actual departure process is associated with departure instants where a cell leaves the switching element. The attempted departure process involves all instants of service completion independent of whether the cell leaves or gets blocked and is forced to receive another service. In this study, we work with the attempted departure process, which we approximate by an IBP. This departure process becomes the offered arrival process to the input buffer of the switching element in the next stage. Part of this process is lost due to cells arriving to a full input buffer.

In order to analyze a switching element in isolation we also require knowledge of the blocking probability P_b. This is the probability that a cell at the head of an output buffer is blocked at the beginning of a switch cycle due to the destination input buffer being full. This probability is used to represent the service at an output buffer by a geometric distribution. That is, a cell receives a service equal to a switch cycle and then with probability $1-P_b$ it departs from the switching element, or with probability P_b it is blocked and it is forced to receive another service.

The attempted departure process from an output buffer and the blocking probability P_b are calculated below in section 3.1. Now, let us assume for a moment that the arrival process to each input buffer of a switching element.and P_b are known. Then, we can analyze a switching element numerically as follows. We first generate all the states of the system, and subsequently we generate the rate matrix $\mathbf{Q}$. The stationary probability vector $\mathbf{x}$ is then obtained by solving the system of linear equations $\mathbf{xQ=0}$.

The state of a switching element is fully described by the variables $\mathbf{n}=\{n_0,n_1,n_2,n_3,n_4\}$. Variable n_0 gives the state of the two arrival processes to the switching element and it takes the following values: 0 if both processes are in the idle state, 1 if the arrival process to the upper input buffer is in the idle state and the arrival process to the lower input buffer is in the active state, 2 if the arrival process to the upper input buffer is in the active state and the arrival process to the lower input buffer is in the idle state, and 3 if both processes are in the active state. Variable n_1 indicates the state of the upper input buffer and it takes the values: 0 if the buffer is empty, 1 if the buffer contains a new cell, 2 if the buffer contains a blocked cell destined for the upper output buffer, and 3 if the buffer contains a blocked cell destined for the lower output buffer. Variable n_2 gives the state of the lower input buffer and it takes the same values as n_1. Finally, variables n_3 and n_4 give the number of waiting cells in the upper output buffer and in the lower output buffer respectively. We have n_3, n_4=0,1,...,N, where N is the capacity of each output buffer. We note that this state description can accommodate two-phase Markov Modulated Bernoulli Processes (MMBP) for the arrival process to an input buffer. Non-uniform destination distributions are achieved by adjusting the branching probabilities r_{uu}, r_{ul}, r_{lu}, and r_{ll}.

The above state description results in $16(N+1)^2+40(N+1)+8$ states. The system of linear equations $\mathbf{xQ=0}$ was solved using Gaussian elimination. Obviously, there are other solution techniques (such as SOR) that are more efficient. However, the particular Gaussian elimination routine that was employed was a very fast routine optimized for a Cray Y-MP. This routine also checked for the stability and reliability of the results.

The buffered Banyan ATM switch is analyzed by decomposing it to the individual switching elements. Each switching element is analyzed numerically as discussed above, given that the attempted departure processes from the corresponding switching elements in the previous stage and the blocking probabilities are known. The parameters of the attempted departure process and the blocking probability are obtained using an iterative scheme. Below, we first obtain the attempted departure process and the blocking probability, and then we describe the iterative scheme.

3.1 The attempted departure process and the blocking probability P_b

In order to characterize the attempted departure process from an output buffer by an IBP, we have to obtain values for p, q, and α. Conceptually, the values for p and q can be obtained as follows. Let us consider the set of all states of the switching element from which it is possible to have an attempted departure from the particular output buffer. If the switching element is in one of these states, then the departure process is in its active state. We shall refer to these states as the *active states*, and their set will be referred to by the symbol *A*. Probability p can then be obtained as the probability that the system will be in one of the active states in the next slot, given that it is currently in an active state. Probability q can be obtained similarly by considering the set of all states of the switching element from which it is not possible to have an attempted departure from the particular output buffer. If the switching element is in one of these states, then the departure process is in its idle state. We shall refer to these states as the *idle states*, and their set will be referred to by the symbol *I*. In view of this, the characterization of the departure process from a particular output buffer involves the classification of the states of the switching element into active and idle states. However, this is not a straight forward task. For example, if α is set so that $\alpha<1$, then certain idle states will have to be classified as active states. Below, we present three different ways for characterizing the departure process as an IBP.

3.1.1 Model 1

In model 1 we simplify the problem of selecting appropriate active and idle states by setting $\alpha=1$. Thus, any state of the switching element from which it is possible to have an attempted departure from the output buffer under consideration is an active state, and all the others are idle states. Let us consider the attempted departure process from the upper output buffer. Then, the set of active states, *A*, consists of any state $\{n_0,n_1,n_2,n_3,n_4\}$ where $n_3>0$. Also, when $n_3=0$, the states $\{n_0,1,0,0,n_4\}$, $\{n_0,1,1,0,n_4\}$, $\{n_0,0,1,0,n_4\}$ are active states with probability r_{uu}, $1-r_{ul}r_{ll}$, and r_{lu} respectively.

The probabilities p and q can be calculated as follows. Let $p(\mathbf{n})$ be the steady-state probability that the switching element is in state $\mathbf{n}$, and $t(\mathbf{n}\rightarrow\mathbf{n}')$ be the transition probability of state $\mathbf{n}$ to state $\mathbf{n}'$. Then, we have

$$p = \frac{\sum_{\mathbf{n} \in A} p(\mathbf{n}) \left(\sum_{\mathbf{n}' \in A} t(\mathbf{n} \rightarrow \mathbf{n}') \right)}{\sum_{\mathbf{n} \in A} p(\mathbf{n})}$$

and

$$q = \frac{\sum_{\mathbf{n} \in I} p(\mathbf{n}) \left(\sum_{\mathbf{n}' \in I} t(\mathbf{n} \rightarrow \mathbf{n}') \right)}{\sum_{\mathbf{n} \in I} p(\mathbf{n})}$$

The problem with model 1 occurs when the input traffic to the switch is bursty. Typically, as bursty traffic proceeds through the Banyan switch, the traffic will slowly lose its burstiness. The squared coefficient of variation C^2 of the inter-arrival time can be seen as a measure of burstiness (see [13]). Comparing the results from model 1 with simulation data, it was observed that the C^2 of the attempted departure process drops off to a very small value immediately after the first stage. This results in a poor estimation of the cell loss at the input of the Banyan switch. Intuitively speaking, this is due to the following situation. Given two IBP arrival processes to a switching element, the output buffer under consideration could be seen as being in the following three distinct states. When both arrival processes to the switching element are idle, the output buffer will become empty, if it has the time to propagate any remaining cells. If one arrival process is active, then the arriving cells will be distributed between the two output buffers. During this time, each output buffer is likely to see very short idle and busy periods. Finally, if both arrival processes are active, it is likely that each output buffer will see substantial traffic, and therefore, its busy period will be long. The active and idle states of the attempted departure process are related to the busy and idle period of the output buffer. The effect of model 1 is that it averages out the busy and idle periods of the output buffer. This results in shorter idle and busy periods of the output buffer, which in turn results in shorter active and idle periods of the attempted departure process. In view of this, the squared coefficient of variation of the attempted inter-departure time drops off substantially. This observation suggests a new model for characterizing the attempted departure process which does not average out all the busy and idle periods of the output buffer. This is be achieved by letting $\alpha < 1$.

The blocking probability P_b can be directly obtained from the steady-state probabilities $p(\mathbf{n})$. We have

$$P_b = \text{Prob}\{\text{input buffer is blocked} \mid \text{attempted arrival}\}.$$

or,

$$P_b = \frac{\sum_{\mathbf{n}\in A} p(\mathbf{n}) \left(\sum_{\mathbf{n}'\in B} t(\mathbf{n}\rightarrow\mathbf{n}') \right)}{\sum_{\mathbf{n}\in A} p(\mathbf{n})}.$$

where B is the set of all states in which the particular input buffer is blocked.

3.1.2 Model 2

The idle states are defined to be the states in which the output buffer is empty, both inputs buffers are empty, and both arrival processes to the switching element are in the idle state. The active states are all the other states. Since it is not possible to have an attempted departure from certain active states, we have that $\alpha < 1$. The length of the time that the departure process is in its active state is longer than in model 1. This is because the set of active states includes all the active states as defined in model 1, plus any state in which at least one of the arrival processes is in the active state. That is, for the departure process from the upper output buffer, the active states are: $\{n_0,n_1,n_2,n_3,n_4\}$, where $n_3>0$, $\{n_0,1,0,0,n_4\}$, $\{n_0,1,1,0,n_4\}$, $\{n_0,0,1,0,n_4\}$ with probability r_{uu}, $1-r_{ul}r_{ll}$, and r_{lu} respectively, and $\{1,0,0,0,n_4\}$, $\{2,0,0,0,n_4\}$, $\{3,0,0,0,n_4\}$ with probability r_{lu}, r_{uu}, and $1-r_{ul}r_{ll}$ respectively.

Let p, q, and α be the parameters of the attempted departure process from an output buffer, say the upper one. Also, let P_b be the blocking probability associated with this output buffer. Then, p, q, and P_b can be obtained in the same way as in model 1. Since there is no cell loss within the switch, α can be obtained by equating throughputs at the output and input of the switching element. Let p_u, q_u, α_u and p_l, q_l, α_l be the parameters of the arrival process to the upper and lower input buffer respectively. These arrival processes are the attempted departure processes from the corresponding output buffers in

the previous stage. Let P_b^u and P_b^l be the blocking probabilities associated with these two (previous stage) output buffers. Then, we have

$$\alpha = \frac{2 \quad p \quad q}{(1 \quad P_b)T(1 \quad q)},$$

where

$$T = \frac{r_{uu}\alpha_u(1 \quad P_b^u)(1 \quad q_u)}{2 \quad p_u \quad q_u} + \frac{r_{lu}\alpha_l(1 \quad P_b^l)(1 - q_l)}{2 - p_l \quad q_l}.$$

Using this new definition of idle and active states, we obtain a model which accurately estimates the squared coefficient of variation C^2 of the time between two successive attempted departures. The accuracy of this model is demonstrated in figure 3 for a Banyan switch consisting of four stages. All arrival processes to the Banyan switch were assumed to be identical and equal to the same IBP process with ρ=0.5. A uniform destination distribution was assumed, i.e. $r_{uu}=r_{ul}=r_{lu}=r_{ll}=0.5$, and the capacity N of each output buffer was 4. The results show analytic and simulation values for the squared coefficient of variation of the time between two actual departures at switch stage 1 through 4. These results were obtained for four different values of the C^2 for the arrival process that actually entered the switch (shown in the graph as stage 0). We note that the analytic results follow the simulation results very closely.

Although model 2 estimates accurately the C^2 of the attempted inter-departure time, it was observed that the value of α was low when compared to simulation results. In model 3, we obtain an IBP where α has a value higher than in the case of model 2.

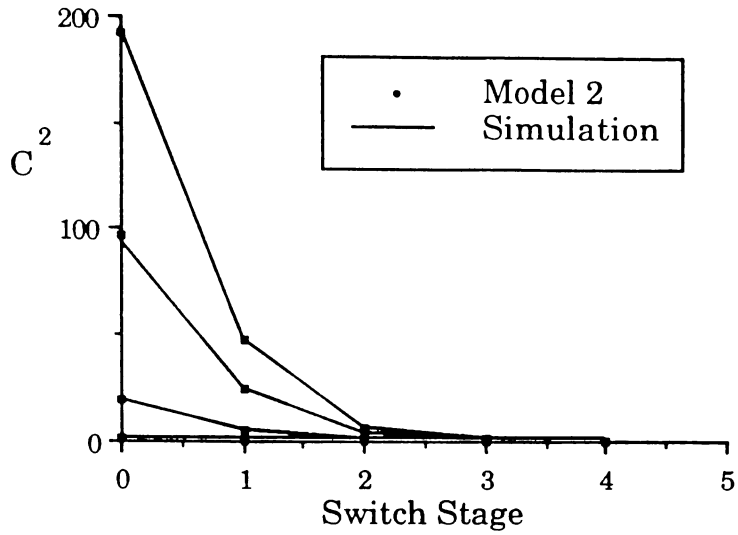

Figure 3: C^2 vs Switch Stage

3.1.3 Model 3

In this model, we start with the parameters obtained from model 2. That is, the definition of idle and active states and the calculation of p, q, α, and P_b are the same as in model 2. From p, q, and α we can obtain the average arrival rate ρ and the squared coefficient of variation C^2. Given ρ and C^2 we specify a new IBP with parameters p', q', and α', where α' is set equal to one, and p' and q' are obtained using ρ and C^2. We have

$$p' = 1 - \frac{2}{C^2(k+1)^2 + k + 1} ,$$

and

$$q' = 1 - kp' , \quad \text{where } k = \frac{\alpha}{1 - \alpha} .$$

Note that model 3 estimates C^2 in the same way as model 2. Setting $\alpha' = 1$ is an arbitrary choice. It has been observed that model 3 provides in most cases an upper bound for the performance measures of interest. In reality, a value for the peak rate between the values obtained by models 2 and 3, seem to be the best choice. Unfortunately there does not appear to be any easy way of obtaining this value, as it depends on the traffic parameters and the topology of the switching element.

3.2 The approximation algorithm

The switch is analyzed by decomposing it into the individual switching elements. Each switching element is then analyzed individually assuming that the arrival process to each input buffer and the blocking probability are known. The entire switch is analyzed using the following iterative scheme. An iteration consists of visiting each switching element beginning at the first stage of the switch and continuing sequentially to the last stage of the switch. For the analysis of each switching element, we use the blocking probabilities from the previous iteration and the attempted arrival processes from the current iteration.

Step 0: For each switching element assume initial values for the blocking probability for each input buffer. Set i to the first stage of the switch (i=1).

Step 1: Analyze numerically each switching element in stage i.

Step 2: For each switching element in stage i calculate the new attempted departure process from each output buffer and the new blocking probability for each input buffer using any of the three models described in section 3.1. Note that the new attempted departure process becomes the input process to the next stage.

Step 3: If stage i is not the last stage of the switch, then set i=i+1 and go to step 1. Else, this iteration is completed. Go to step 4.

Step 4: The convergence test compares $\mathbf{p(n)}$ for each switching element from one iteration to the next. The tolerance we used was nine decimal place accuracy. Stop, if convergence has occurred. Else set i=1 and go to step 1.

An alternative iterative scheme that can be employed to analyze the switch is the scheme that is commonly used to analyze tandem configurations of open queueing networks with finite capacity queues. This scheme consists of a forward and a backward pass. In the forward pass, the switching elements are analyzed from the first stage to the last stage and the attempted departure process from each output buffer is updated. In the backward pass, the switching elements are analyzed from the last stage back to the first stage and the blocking probabilities are updated. This scheme has been found empirically to require more CPU time than the scheme presented above.

4. Results

The approximation model described in the previous section was implemented on a Cray Y-MP. All three ways of characterizing the attempted departure process from the output buffer of a switching element were implemented. We shall refer to the approximation

model as model 1, 2, or 3, depending upon which model of the attempted departure process was used. The main performance measures obtained by the model were the cell loss probability that occurs at the input of the Banyan switch, the mean time to traverse the switch, hereafter referred to as *mean delay*, and the throughput which is defined as the probability that a slot is busy. Queue length distributions within a switch were also captured but they were not reported seeing that they are reflected in the above three measures, and in particular in the mean delay. Note also that the cell loss is directly related to the throughput.

The approximation results were compared to simulation data. A representative sample of these comparisons is given below for a 16x16 switch. Figures 4 to 15 have been obtained assuming that all the arrival processes to the Banyan switch are identical and equal to the same IBP process. Also, uniform destination distribution has been assumed, i.e. for each switching element $r_{uu}=r_{ul}=r_{lu}=r_{ll}=0.5$. This type of traffic will be referred to as *symmetric traffic*. Figures 16 through 23 have been obtained assuming different arrival processes to the switch and and non-uniform destination distributions. This type of traffic will be referred to as *asymmetric traffic*. In all the examples given below, the capacity N of each output buffer was 4 unless otherwise stated.

Now, let us consider figures 4 to 15. Let ρ and C^2 be the average arrival rate and the squared coefficient of variation of the inter-arrival time of the common IBP that characterizes each input source to the switch. Figures 4 and 6 report the accuracy of model 3 in estimating the cell loss probability. Note that the vertical bars show the confidence intervals associated with the simulation results. (In general, we will not show confidence intervals in a graph, if they are very small.) Figures 5 and 7 provide the respective absolute error. Absolute error was used for cell loss probabilities due to the exceptionally small values. Models 1 and 2 do not provide as accurate an estimation of the cell loss probability.

Figures 8 and 10 provide mean delay results. The respective relative errors are provided in figures 9 and 11. Model 1 provides the best accuracy in this case. Poorer accuracy is attained as the traffic becomes more bursty.

Figure 12 offers throughput estimations for a wide range of ρ and varying C^2. The relative errors for the worst case curve ($\rho = 0.9$) are given in figure 13. All three models perform extremely well in regards to throughput. Figure 14 shows the affect of output

buffer size on throughput . The relative errors provided in figure 15 shows the accuracy of model 3 over different buffer sizes.

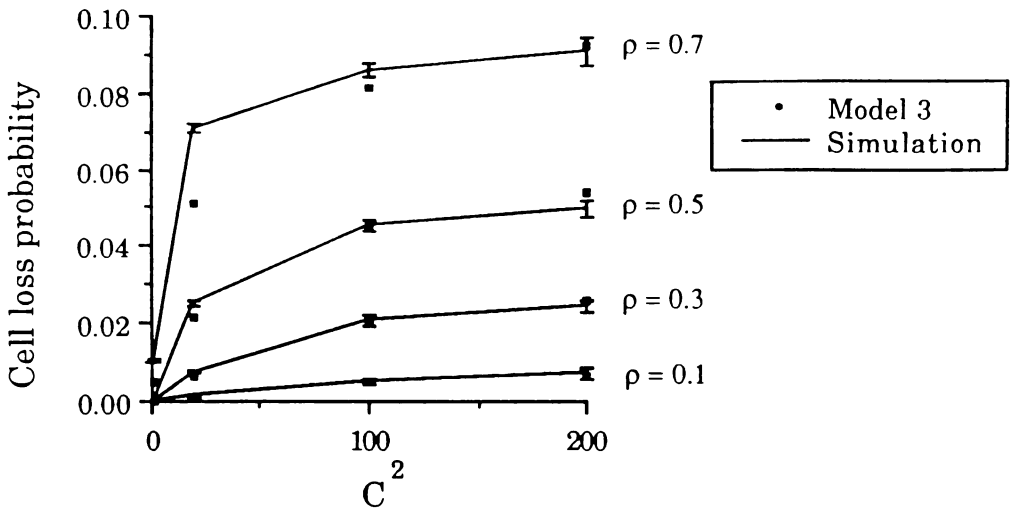

Figure 4: Cell loss probability vs C^2

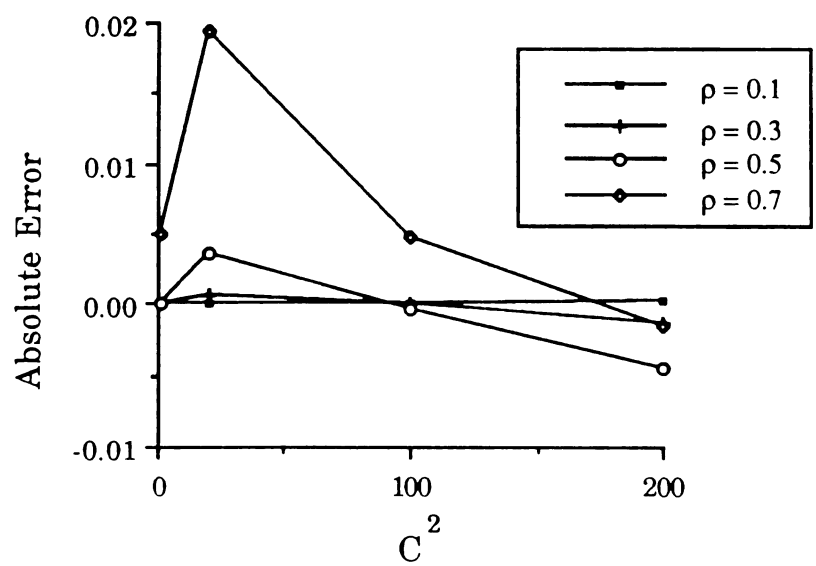

Figure 5: Absolute Error for results in figure 4

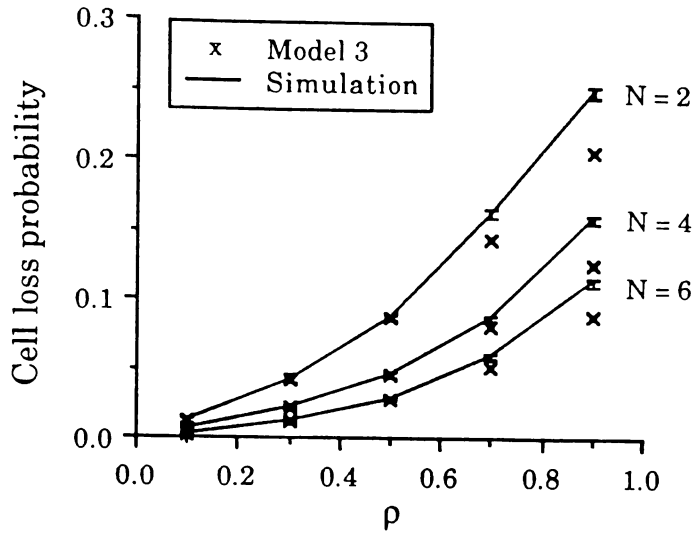

Figure 6: Cell loss probability vs ρ ($C^2 = 100$)

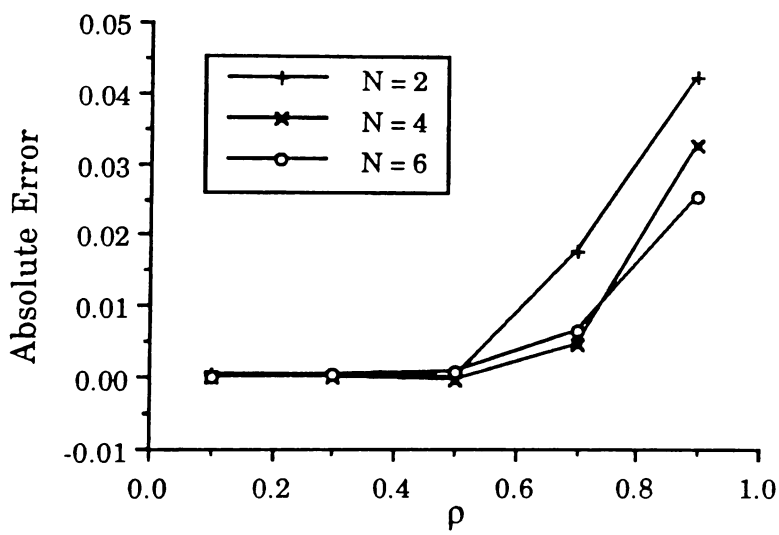

Figure 7: Absolute Error for results in figure 6

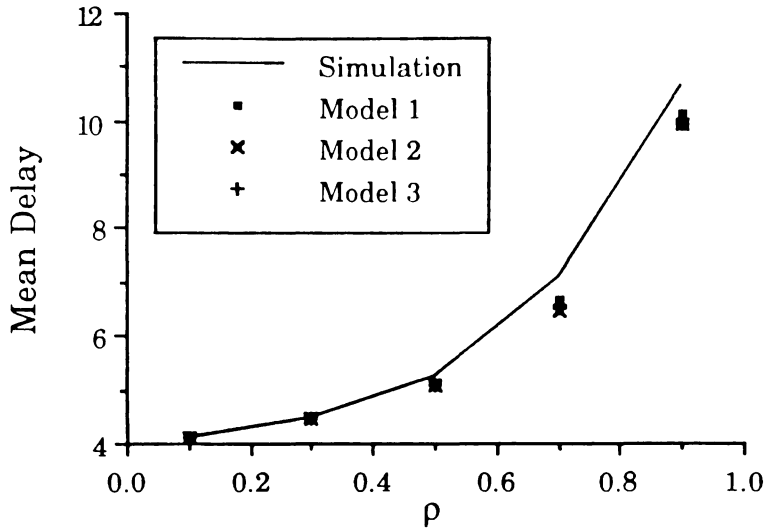

Figure 8: Mean Delay vs ρ ($C^2 = 1$)

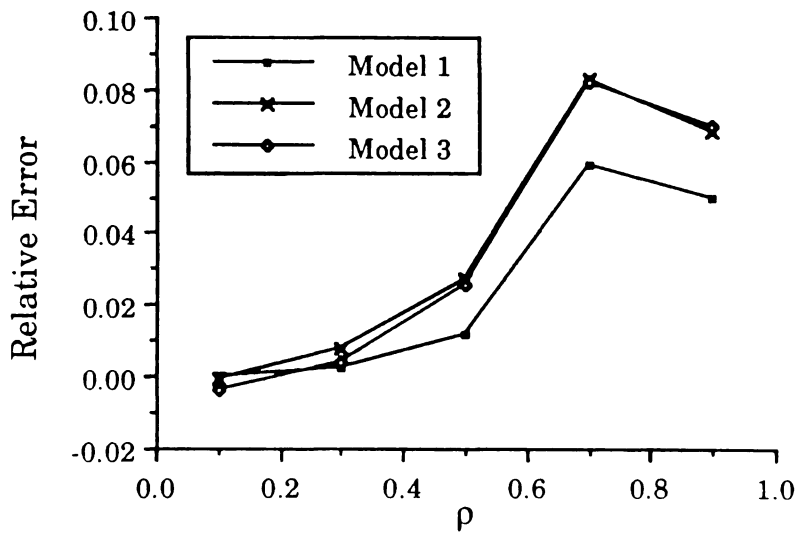

Figure 9: Relative Error for results in figure 8

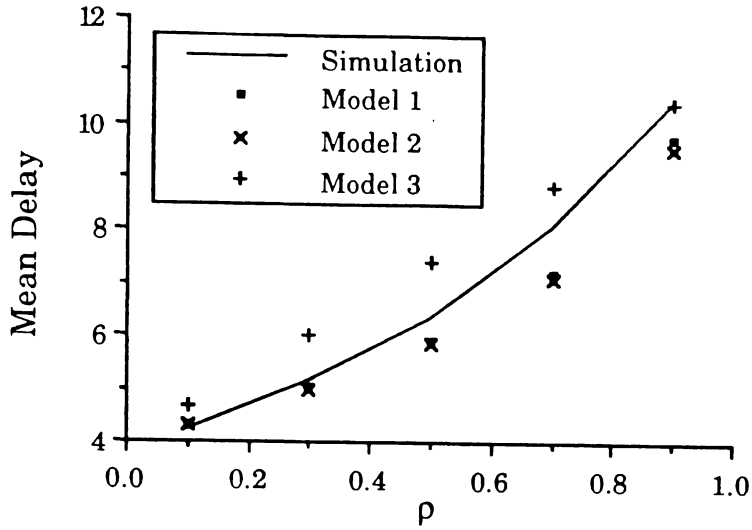

Figure 10: Mean Delay vs ρ ($C^2 = 200$)

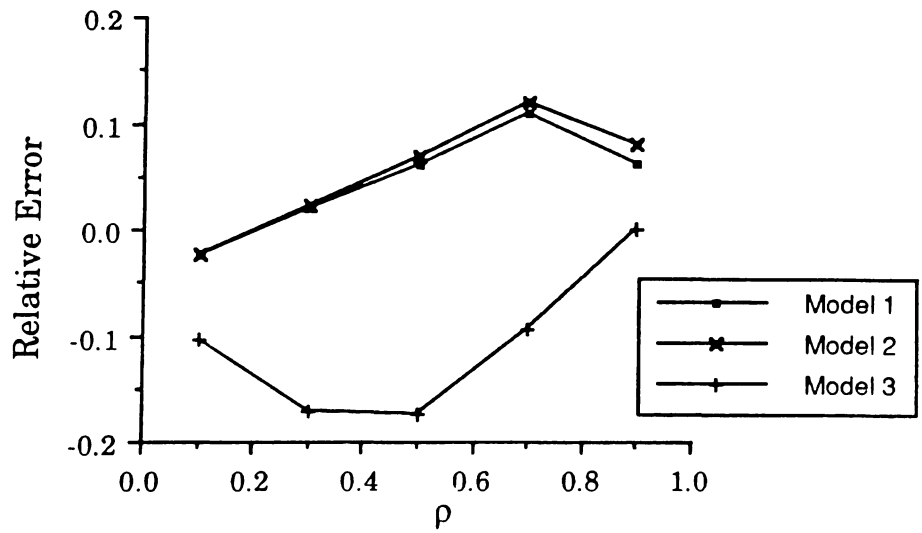

Figure 11: Relative Error for results in figure 10

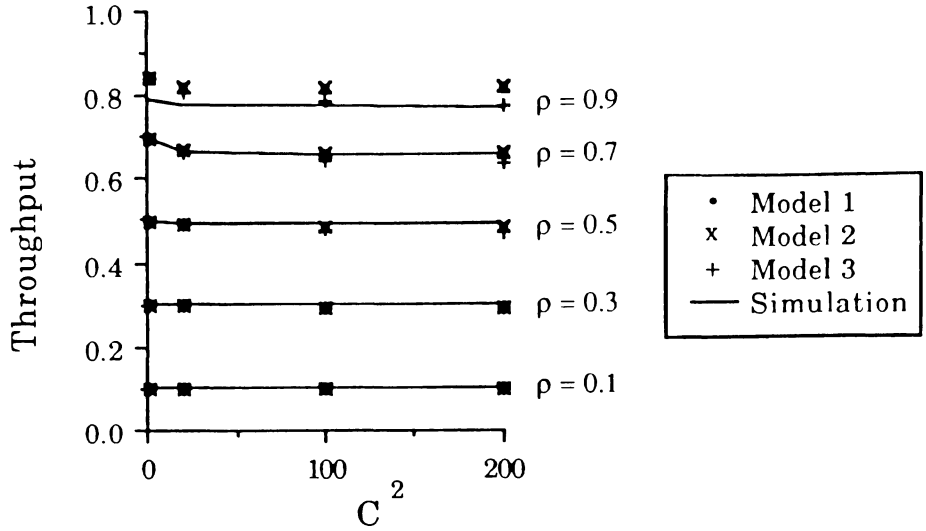

Figure 12: Throughput vs C^2

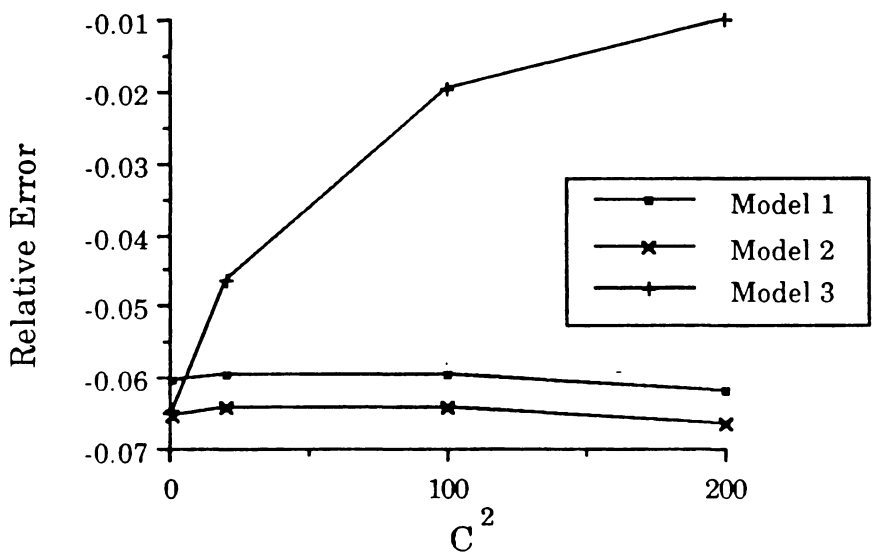

Figure 13: Relative Error for worst-case results in figure 12 ($\rho = 0.9$)

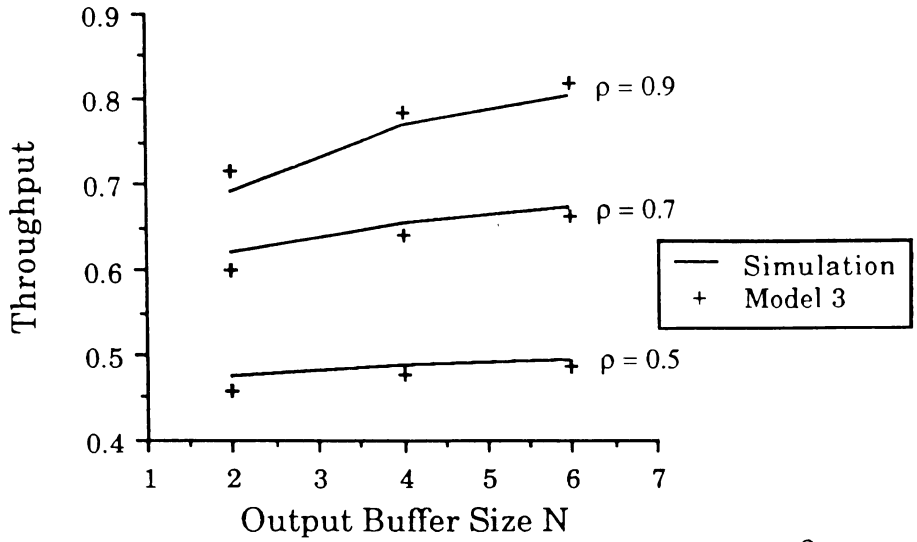

Figure 14: Throughput vs Output Buffer Size N ($C^2 = 100$)

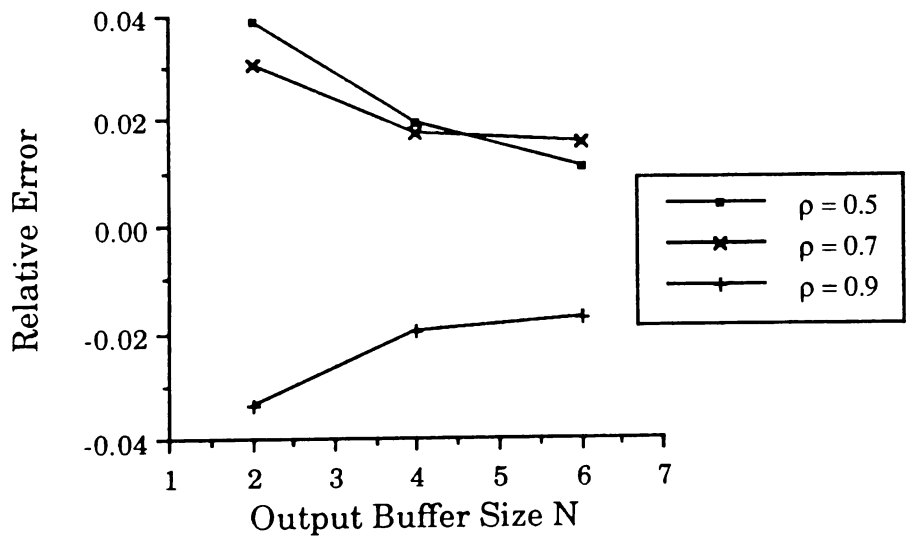

Figure 15: Relative Error for results in figure 14

The most severe traffic pattern that provides the highest blocking and the widest variation of traffic across the switch is the hotspot pattern (see [14], [15]). This asymmetric type of traffic is implemented as follows. We assume that all the arrival processes at the input ports of the switch are identical and equal to the same IBP. However, a percent of each input source is destined for the same output (hereafter referred to as the *hot output*). The remainder of the traffic is distributed uniformly amongst all destinations including the hot output. The output which receives the least amount of traffic is defined as the *non-hot output*. Let ρ and C^2 be the average arrival rate and the squared coefficient of variation of the inter-arrival time of the common IBP that represents all input sources to the switch. Figures 16 and 18 provide the throughput as a function of the hotspot percentage for two different values of ρ. We note that both models 2 and 3 have very good accuracy, even when the throughput is very close to one. Similar results were obtained for model 1. The respective relative errors are given in figures 17 and 19. It is also interesting to see how well the models estimate the throughput within the switch itself. Correct internal approximation will result in an accurate model regardless of switch size. Figure 20 shows how well models 2 and 3 estimate the throughput within each stage of the Banyan switch even with bursty traffic. The relative error is provided in figure 21.

We now consider the case where each input source to the switch is different. Let us number sequentially the input sources of a 16x16 switch such that the uppermost source is 1, and the lowest source is 16. Using this notation, C^2 was varied from 160 to 10. The highest burstiness was provided for source 1, and the least for source 16. Each source i had a C^2 of 10 less than that of the source i-1. The average bandwidth ρ was also varied. Source 1 offered a ρ=0.9, and source 16 a ρ=0.15, with any source i having a ρ of 0.05 less than source i-1. A uniform destination distribution was used. Figure 22 shows the throughput at each stage within the switch. The top curve shows the uppermost path (source 1 to output 1) and the bottom curve is the lowest path (source 16 to output 16). The model estimates the throughput within the switch very accurately. Due to the uniform destination distribution, the two curves converge since each output of the switch behaves identically. The relative error is provided in figure 23.

Note that the cell loss probability and the mean delay was not presented for asymmetric traffic. However, the accuracy of these performance measures under asymmetric traffic has been found to be similar to that of the symmetric traffic case.

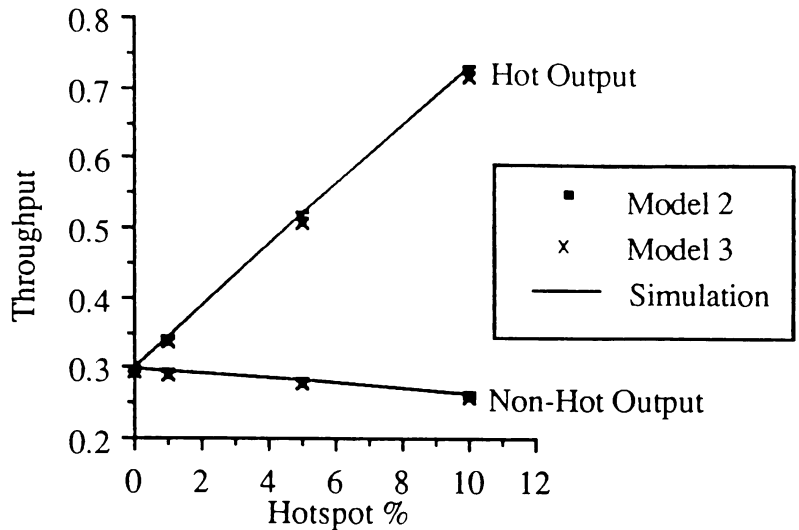

Figure 16: Throughput vs Hotspot % ($C^2 = 200, \rho = 0.3$)

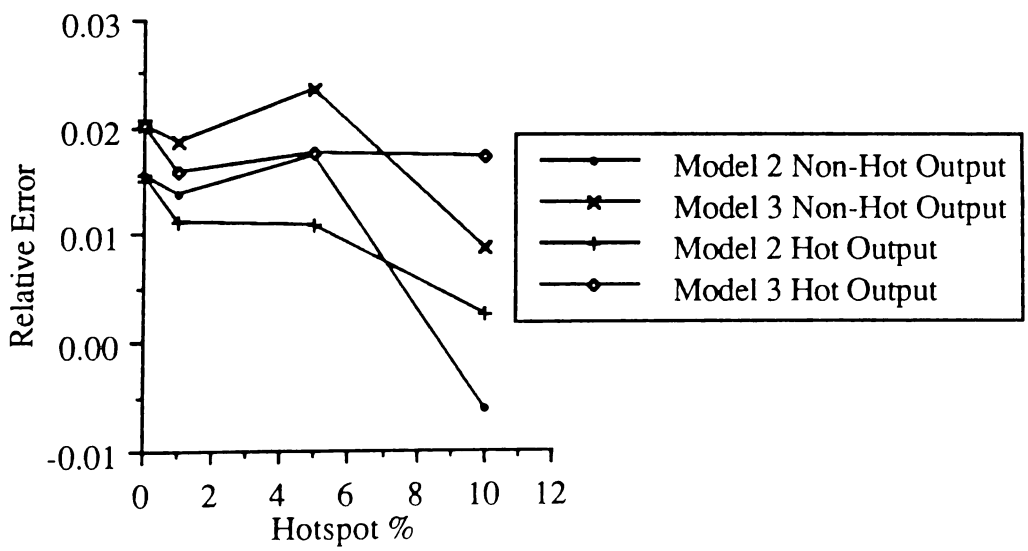

Figure 17: Relative Error for results in figure 16

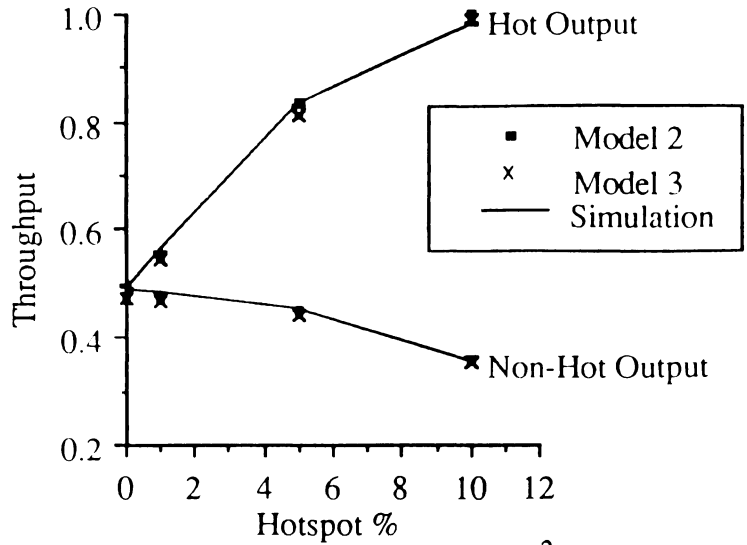

Figure 18: Throughput vs Hotspot % ($C^2 = 200, \rho = 0.5$)

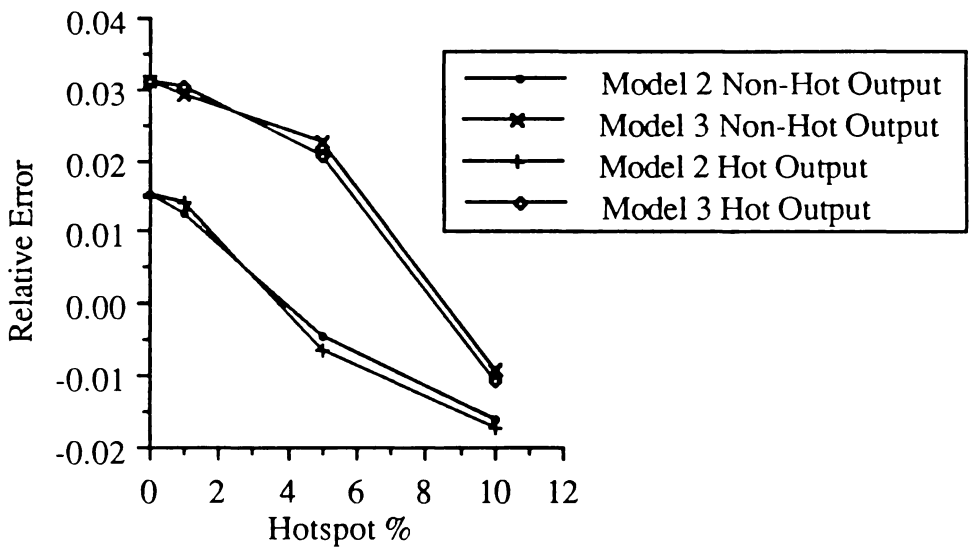

Figure 19: Relative Error for results in figure 18

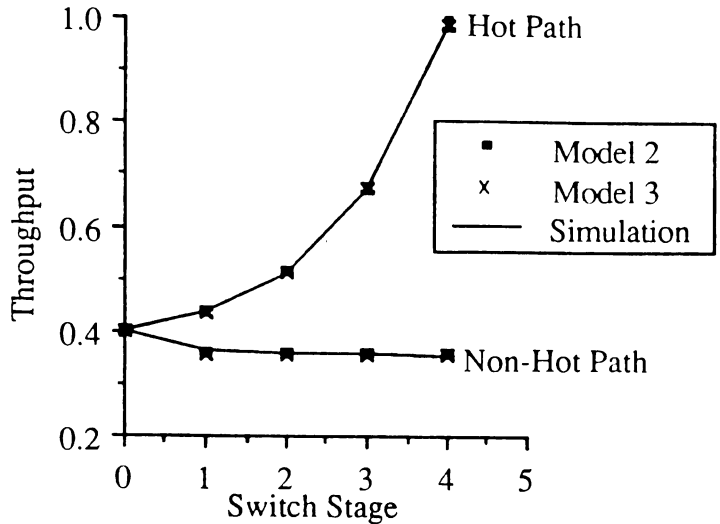

Figure 20: Throughput vs Switch Stage
(C^2 = 200, ρ = 0.5, Hotspot % = 10)

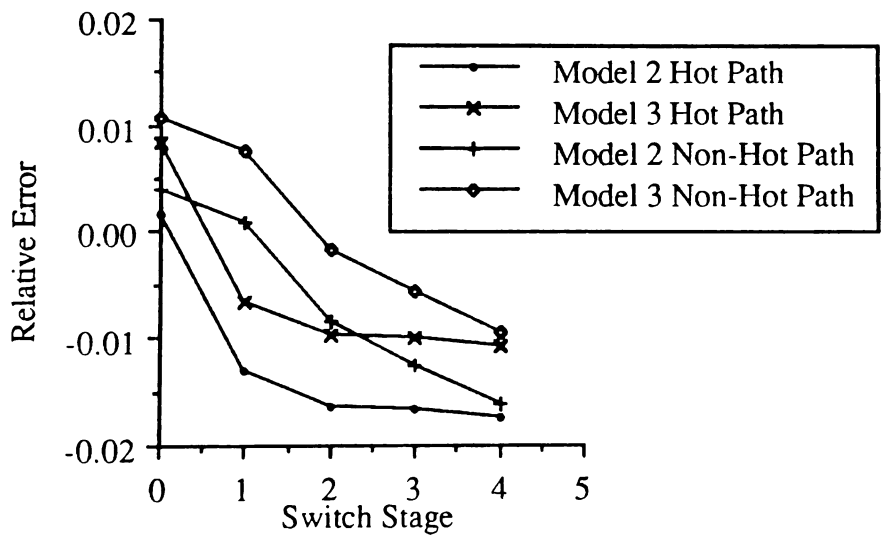

Figure 21: Relative Error for results in figure 20

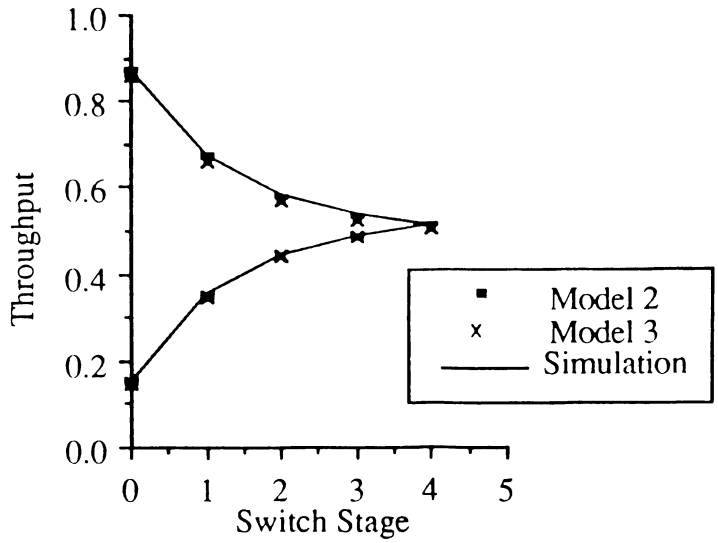

Figure 22: Throughput vs Switch Stage

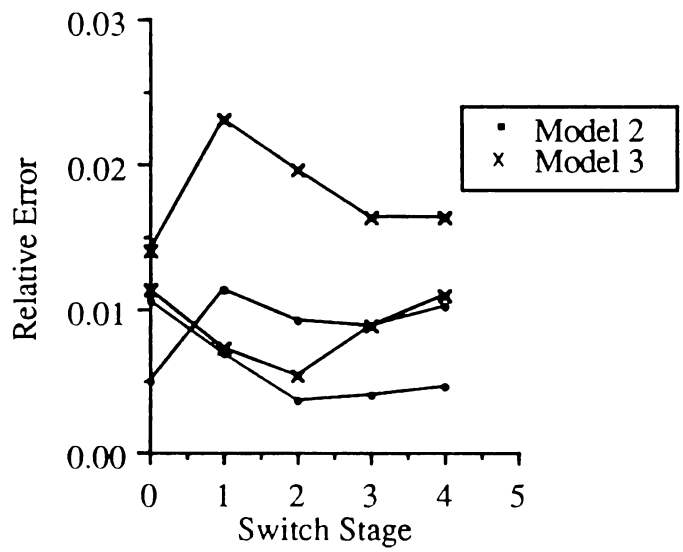

Figure 23: Relative Error for results in figure 22

In general, all three models estimate the throughput with very little error. Model 1 tends to provide a better estimate of the mean delay, whereas model 3 provides better estimates of the cell loss probability.

The analytical model consumed roughly 10 to 500 times less computer time than the equivalent simulation. The CPU time was primarily dependent on C^2, ρ, and N. Run times for the symmetric traffic examples ranged from 6 to 240 seconds, whereas for the asymmetric traffic examples 5 to 15 minutes was required. All experiments were carried out on a Cray Y-MP. Convergence of the iterative algorithm generally occurred between the fifth and fifteenth iteration. The difference between the run times for the symmetric and asymmetric traffic examples is due to the fact that in the symmetric traffic examples only one row of switching elements was analyzed. In the asymmetric traffic examples, all 32 switching elements were analyzed. In the symmetric examples, each iteration consisted in analyzing 4 switching elements, whereas in the asymmetric examples each iteration consisted of analyzing 32 switching elements.

Figure 24 shows the C^2 of the actual departure process from each switch stage in a very large switch (1048576 x 1048576, ie. 20 stages). Symmetric traffic was assumed. The C^2 for the arrival process that actually entered the switch is shown as stage 0. Note that C^2 eventually becomes equal to $1-\rho$, which is the C^2 of a Bernoulli process. In fact, in the analytical model, it was observed that the attempted departure process becomes a Bernoulli process after about 10 switch stages since $p \rightarrow 1$ and $q \rightarrow 0$. Our extensive simulations have shown that the actual departure process from a switch stage is not Bernoulli, even if the offered arrival process to the switch is Bernoulli. The reason for this is the back-pressure blocking mechanism. As discussed previously, when blocking occurs the blocked output queue tends to grow. After blocking, the input buffer is not exposed to a Bernoulli source but rather a number of attempted arrivals which will eventually become a Bernoulli source (assuming that no further blocking occurs). Previous analytic models which have been based on Bernoulli arrival and departure processes have been shown to perform poorly at high offered loads where blocking occurs. Our model performs better at high offered loads due to the IBP characterization of the attempted departure process.

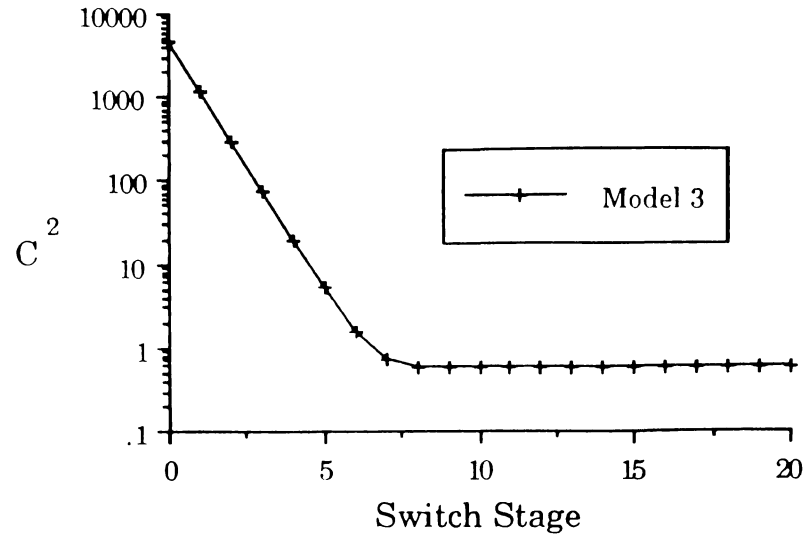

Figure 24: C^2 vs Switch Stage

(ρ = 0.5, Switch Size is 1048576 x 1048576)

5. Conclusions

In this paper, we presented an analytical model of a buffered Banyan ATM switch. The model allows complex switching elements, bursty traffic, and non-uniform destination distributions and permits the analysis of large scale switches. The ATM switch was analyzed by decomposing it into individual switching elements. Each switching element was then analyzed numerically in isolation assuming that its arrival and service processes are known. Three different methods for characterizing the arrival process to an input buffer of a switching element were presented. The parameters of the arrival and service processes of the switching elements were obtained using an iterative scheme. The results obtained are approximate and validation tests have shown that they have good accuracy. Using this model, the cell loss, throughput, and the mean time to traverse the switch were obtained for different traffic parameters, and buffer sizes within a switching element.

The characterization of the attempted departure process by an IBP provides acceptable accuracy. In fact the accuracy of the model far surpasses previous models of this nature. A more accurate characterization of the attempted departure process would likely require a three-phase Markov Modulated Bernoulli Process, representing an idle

state, a medium arrival rate state, and a high arrival rate state. The three-phase MMBP, however, would greatly increase the state space of a switching element. A non-Markovian process could also be used. However, given that the current model has a satisfactory accuracy, it is probably not necessary to develop arrival processes which are more complex. In fact, a reduction of the state space of the switching element is more desirable. One possible reduction is the removal of state information which specifies the address of the output buffer of a blocked input buffer.

Acknowledgements

The authors wish to thank the North Carolina Supercomputing Center for the use of their Cray Y-MP and IBM 3090 computers.

References

[1] S. Sumita and T. Ozawa, "Achievability of performance objectives in ATM switching nodes", in Symposium on Performance of Distributed and Parallel Systems, 1988, pp. 45-56, Kyoto, Japan.

[2] K. Shiomoto, M. Murata, Y. Oie, and H. Miyahara, "Performance evaluation of cell bypass queueing discipline for buffered Banyan type ATM switches", in Proceedings of INFOCOM '90, pp. 677-685.

[3] Y. Liu and S. Dickey, "Simulation and analysis of different switch architectures for interconnection networks in MIMD shared memory machines", Ultracomputer Note # 141, 1988.

[4] S. Dickey, A. Gottlieb, R. Kenner, and Y.liu, "Designing VLSI network nodes to reduce memory traffic in a shared memory parallel computer, Ultracomputer Note # 125, 1986.

[5] G. Pfister, W. Brantley, D. George, S. Harvey, W. Kleinfelder, K. McAuliffe, E. Melton, V. Norton, and J. Weiss, "The IBM research parallel processor prototype (RP3): introduction and architecture", In Proceedings of the 1985 International Conference on Parallel Processing, pp. 764-771, St. Charles, Il., Aug. 20-23, 1985.

[6] J.H. Patel, "Performance of processor-memory interconnections for multiprocessors", IEEE Trans. on Computers, C-30(10), pp. 771-780, Oct. 1981.

[7] Y. Jenq, "Performance analysis of a packet switch based on single-buffered Banyan network", IEEE Journal on Selected Areas of Communication, vol. SAC-1, pp. 1014-1021, Dec. 1983.

[8] A. Saha and M. Wagh, "Performance analysis of Banyan networks based on buffers of various sizes", in Proceedings of INFOCOM '90, pp. 157-163.

[9] T. Theimer, E. Rathgeb, and M. Huber, "Performance analysis of buffered Banyan networks", in Symposium on Performance of Distributed and Parallel Systems, 1988, pp. 57-72, Kyoto, Japan.

[10] H. Kim and A. Leon-Garcia, "Performance of buffered Banyan networks under nonuniform traffic patterns", IEEE Transactions on Communications, vol. 38, no. 5, pp. 648-658, May 1990.

[11] T. Eliazov, V. Ramaswami, W. Willinger, and G. Latouche, "Performance of an ATM switch: simulation study", in Proceedings of INFOCOM '90, pp. 644-660.

[12] H. Kim and A. Leon-Garcia, "Performance of self-routing ATM switch under nonuniform traffic pattern", in Proceedings of INFOCOM '90, pp. 140-147.

[13] A.A. Nilsson, F.-Y. Lai, H.G. Perros, "An approximate analysis of a bufferless NxN synchronous Clos ATM switch", to appear ITC-13.

[14] H.G. Perros, "Approximation algorithms for open queueing networks with blocking", in Takagi (Ed.) *Stochastic Analysis of Computer and Communication Systems* (North-Holland, 1989).

[15] G. Pfister and V. Norton, "Hot spot contention and combining in multistage interconnection networks", IEEE Trans. on Computers, vol. c-34, no. 10, pp. 943-948, Oct. 1985.

[16] T. Lang and L. Kurisaki, "Nonuniform traffic spots (NUTS) in multistage interconnection networks", In Proceedings of the 1988 International Conf. on Parallel Processing, vol I: Architecture, pp. 191-195, St. Charles, Il., Aug. 1988.