ANALYTICAL ANALYSIS OF ATM SWITCHES WITH MULTIPLE INPUT QUEUES WITH BURSTY TRAFFIC*

Ge Nong, Mounir Hamdi and Jogesh K. Muppala
Department of Computer Science
The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
Email: nong@ieee.org and {hamdi,muppala}@cs.ust.hk

Abstract

A queueing model for a novel multiple input-queued ATM switch under *i.i.d* bursty traffic modeled by 2-state *Markov Modulated Bernoulli Processes* (MMBPs) is proposed. A *Quasi-Birth-Death* (QBD) chain is constructed as the underlying Markov chain of the queueing model. Each input port of the switch maintains N separate queues each for buffering cells destined to one of the N outputs and an efficient randomized parallel algorithm, called *parallel iterative matching* (PIM) is used by the switch to schedule the head-of-line (HOL) cells of the input queues out to the output queues. The QBD chain is solved by finding the *fixed point* of the introduced *fixed point equation* using an iterative computing scheme. Interesting performance parameters of the switch such as the throughput, the mean cell delay and the cell loss probability are derived from the solved QBD chain. Numerical results from both the analytical model and simulations are presented and the accuracy of the analysis is discussed. The queueing model can be extended using the same technique to the situation where complicated bursty traffic with more states is asserted to the switch.

1 Introduction

Each input of an ATM switch scheduled by the PIM algorithm maintains N separate queues each for cells destined for one of the N outputs. The switch operates synchronously and in each time slot the *head-of-line* (HOL) cells at the input queues can be selected for transmission across the switch with the constraint that at most one cell is able to go from/to any one input/output link. The performance evaluations of PIM switches found in the literature were all based on simulations except in [6, 7], where analytical models were constructed to study the performance of PIM switches under *i.i.d Bernoulli* traffics and the accuracy of the analytical models were verified by simulations. Unfortunately, most network traffic are known to be bursty rather than Bernoulli. As a result, in this paper, we develop an analytical model for a PIM switch under *i.i.d* bursty traffics modeled by *i.i.d* 2-state *Markov-modulated Bernoulli Processes* (MMBPs) [1]. Numerical results show that our analytical model work quite well and can be used as an efficient tool to evaluate various performance parameters of the PIM switch such as the cell loss probability which can be very time-consuming and sometimes impossible to obtain by simulations.

The remainder of this paper is organized as follows. In Section 2 the queueing model is proposed for a PIM switch with bursty traffic and a QBD underlying Markov chain is constructed. In addition, the QBD chain is solved by a *fixed point* iterative method. In Section 3 comparisons of the numerical results from the queueing model with the results of simulations are presented. Finally, a conclusion is given in Section 4.

2 Queueing Model and Analysis of the PIM Switch

For sake of simplicity and clarity, we apply the analysis in this paper to a modified PIM algorithm which is **logically equivalent** to the original PIM algorithm, instead of using the original PIM algorithm directly. The detailed descriptions for the original PIM scheduling algorithm and its modified logically equivalent counterpart as well as the proof of their logical equivalence for our purpose can be found in [2, 7]. In

0-7803-5796-5/99/$10.00 © 1999 IEEE

essentially the same in the context of this paper.

2.1 Queueing Model

A number of *assumptions* are made for developing the queueing model of the PIM switch: (1) The switch operates synchronously; (2) Every input queue has the same buffer size, namely b_i; (3) New cells arrive only at the beginning of the time slots and cells depart only at the end of the time slots; and (4) Cells arrive at each input according to an ON-OFF bursty process [1] modeled by the 2-state MMBP. In this process cells are only generated in ON(1) state and the destinations of cells are uniformly distributed over all outputs. Only one cell can arrive at each input in a time slot. Times spent in states of ON(1) and OFF(0) are geometric with means of $(1-\alpha)^{-1}$ and $(1-\beta)^{-1}$, respectively. For an $N \times N$ switch, if an input's load is $N\lambda$, then every queue at this input has an offered load of λ. Given the mean burst length τ and the mean arrival rate λ, α and β can be calculated as $\alpha = 1 - \frac{1}{\tau}$ and $\beta = \frac{1-N\lambda(2-\alpha)}{1-N\lambda}$, respectively.

Holding the above assumptions, all the input queues' stochastic processes will be the same when the system attains equilibrium steady states. We refer to a queue at input i with output j as the destination by $Q(i,j)$. Figure 1 shows an example of the queueing model for the PIM switch. In this example the occupancy of $Q(1,1)$ is taken as the *tagged input queue*, the number of HOL cells at input 1 is represented by the *1st HOL input queue*, and the number of HOL cells addressed for output 1 is denoted by the *1st HOL output queue*. Both the *HOL input queue* and the *HOL output queue* are virtual queues which don't exist in a real PIM switch but are used in our analysis to represent the cells participating in the two stages' contentions of the PIM scheduling algorithm. Without loss of generality, $Q(i,j)$ is assumed to be the *tagged input queue* in the rest of this paper.

2.2 Underlying Markov Chain

The queueing model is analyzed by constructing an underlying Markov chain Z which states are sampled at the end of each time slot and each state is expressed as a 4-tuple (L, G, W_i, W_o), where L, G, W_i, and W_o refer to the length of the *tagged input queue*, the state of the traffic source at the tagged input queue, the length of the *virtual HOL input queue*, and the length of the *virtual HOL output queue*, respectively. The state space of this four-dimensional Markov chain is

$$\{(0,g,0,0),(l,g,w_i,w_o) \mid 1 \le l \le b_i, 0 \le g \le 1, \\ 1 \le w_i \le N, 1 \le w_o \le N\}$$

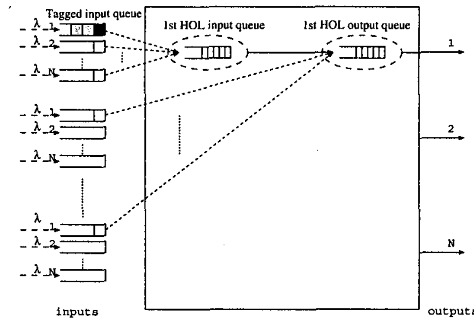


Figure 1: An example of the queueing model for the PIM switch.

and are ordered in a lexicographical order, that is, (0,0,0,0), (0,1,0,0), (1,0,1,1), ..., $(b_i, 0, N, N)$, $(b_i, 1, N, N)$. The set of states $\{(l,0,1,1),(l,1,1,1),...(l,1,2,2),...(l,1,N,N)\}$ will be labeled as states in level l of the Markov chain. The Markov chain Z is a so-called QBD process with a block-partitioned form of transition probability matrix as follows:

$$T = \begin{bmatrix} C_1 & C_2 & 0 & & & & & \\ C_0 & A_1 & A_2 & 0 & & & & \\ 0 & A_0 & A_1 & A_2 & 0 & & & \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & & \\ 0 & 0 & 0 & \cdots & 0 & A_0 & A_1 & A_2 \\ 0 & 0 & 0 & \cdots & 0 & 0 & D_0 & D_1 \end{bmatrix}$$

where $C_1[1,1]^T + C_2 e = [1,1]^T$ and $C_0 + (A_1 + A_2)e = (A_0 + A_1 + A_2)e = (D_0 + D_1)e = e$, e is a column vector of ones of length $2N^2$. Let $P_{blo,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$ ($P'_{blo,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$, resp.) denote the probability that the HOL cell of the queue $Q(i,j)$ is blocked, and $P_{suc,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$ ($P'_{suc,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$, resp.) denote the probability that the HOL cell of the queue $Q(i,j)$ is transmitted given that (i) there is(isn't, resp.) a new cell arrival at the queue $Q(i,j)$ at the beginning of the current time slot; (ii) at the end of the last time slot, the traffic source at input i is in state g and the lengths of the virtual HOL input and output queues are w_i and w_o; (iii) at the end of the current time slot, the lengths of the virtual HOL input and output queues are w_i' and w_o'. For the first case of (i), i.e., there is a new arrival cell at the *tagged input queue* at the beginning of the current time slot, we define six matrices $B^{(g)}$, $B_0^{(g)}$ and $S^{(g)}$ ($g = 0$ or $g = 1$) as:

$$B^{(g)} = \begin{bmatrix} P_{blo,W_t(1,1)|W_{t-1}(g,1,1)} & P_{blo,W_t(1,2)|W_{t-1}(g,1,1)} & \cdots & P_{blo,W_t(N,N)|W_{t-1}(g,1,1)} \\ P_{blo,W_t(1,1)|W_{t-1}(g,1,2)} & P_{blo,W_t(1,2)|W_{t-1}(g,1,2)} & \cdots & P_{blo,W_t(N,N)|W_{t-1}(g,1,2)} \\ P_{blo,W_t(1,1)|W_{t-1}(g,1,3)} & P_{blo,W_t(1,2)|W_{t-1}(g,1,3)} & \cdots & P_{blo,W_t(N,N)|W_{t-1}(g,1,3)} \\ \vdots & \vdots & \cdots & \vdots \\ P_{blo,W_t(1,1)|W_{t-1}(g,N,N)} & P_{blo,W_t(1,2)|W_{t-1}(g,N,N)} & \cdots & P_{blo,W_t(N,N)|W_{t-1}(g,N,N)} \end{bmatrix}$$

$$B_0^{(g)} = [P_{blo,W_t(1,1)|W_{t-1}(g,0,0)},\ P_{blo,W_t(1,2)|W_{t-1}(g,0,0)},\ \cdot s,\ P_{blo,W_t(N,N)|Wt-1(g,0,0)}]$$

$$S^{(g)} = \begin{bmatrix} P_{suc,W_t(1,1)|W_{t-1}(g,1,1)} & P_{suc,W_t(1,2)|W_{t-1}(g,1,1)} & \cdots & P_{suc,W_t(N,N)|W_{t-1}(g,1,1)} \\ P_{suc,W_t(1,1)|W_{t-1}(g,1,2)} & P_{suc,W_t(1,2)|W_{t-1}(g,1,2)} & \cdots & P_{suc,W_t(N,N)|W_{t-1}(g,1,2)} \\ P_{suc,W_t(1,1)|W_{t-1}(g,1,3)} & P_{suc,W_t(1,2)|W_{t-1}(g,1,3)} & \cdots & P_{suc,W_t(N,N)|W_{t-1}(g,1,3)} \\ \vdots & \vdots & \cdots & \vdots \\ P_{suc,W_t(1,1)|W_{t-1}(g,N,N)} & P_{suc,W_t(1,2)|W_{t-1}(g,N,N)} & \cdots & P_{suc,W_t(N,N)|W_{t-1}(g,N,N)} \end{bmatrix}$$

In case there is no new cell arrival at the *tagged input queue* at the beginning of the current time slot, we define another six matrices $B'^{(g)}$, $B_0'^{(g)}$ and $S'^{(g)}$ similar to $B^{(g)}$, $B_0^{(g)}$ and $S^{(g)}$ by replacing $P_{blo,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$ in $B^{(g)}$ with $P'_{blo,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$, $P_{blo,W_t(w_i',w_o')|W_{t-1}(g,0,0)}$ in $B_0^{(g)}$ with $P'_{blo,W_t(w_i',w_o')|W_{t-1}(g,0,0)}$ and $P_{suc,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$ in $S^{(g)}$ with $P'_{suc,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$, respectively. Using the above definitions, the element matrices in the transition probability matrix T can be computed as below:

$$C_0 = \begin{bmatrix} H_0 e_1 & (S'^{(0)} - H_0)e_1 \\ H_1 e_1 & (S'^{(1)} - H_1)e_1 \end{bmatrix},$$

$$C_1 = \begin{bmatrix} \beta & 1-\beta-B_0^{(0)}e_1 \\ 1-\alpha & \alpha - B_0^{(1)}e_1 \end{bmatrix}, \quad C_2 = \begin{bmatrix} z_r & B_0^{(0)} \\ z_r & B_0^{(1)} \end{bmatrix},$$

$$A0 = \begin{bmatrix} H_0 & S'^{(0)} - H_0 \\ H_1 & S'^{(1)} - H_1 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} G_0 & S^{(0)} + B'^{(0)} - G_0 \\ G_1 & S^{(1)} + B'^{(1)} - G_1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} z_m & B^{(0)} \\ z_m & B^{(1)} \end{bmatrix},$$

$$D_0 = \begin{bmatrix} H_0 & S^{(0)} + S'^{(0)} - H_0 \\ H_1 & S^{(1)} + S'^{(1)} - H_1 \end{bmatrix},$$

$$D_1 = \begin{bmatrix} G_0 & B^{(0)} + B'(0) - G_0 \\ G_1 & B^{(1)} + B'(1) - G_1 \end{bmatrix};$$

where e_1 is a column vector of ones of size N^2, z_r/z_c is a row/column vector of zeros of length N^2, z_m is an $N^2 \times N^2$ matrix of zeros and $H_0 = \frac{S^{(0)}\beta}{(1-\beta)/N}$, $H_1 = \frac{S^{(1)}(1-\alpha)}{\alpha/N}$, $G_0 = \frac{B^{(0)}\beta}{(1-\beta)/N}$ and $G_1 = \frac{B^{(1)}(1-\alpha)}{\alpha/N}$.

The remaining subsections will cover the computation of the success and blocking probabilities we defined above, i.e., $P_{suc,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$, $P'_{suc,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$, $P_{blo,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$ and $P'_{blo,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$, respectively. Provided that these probabilities are computed, the transition probability matrix T can be constructed. Once the transition probability matrix is known, it is a routine matter to derive the steady state equations by utilizing the properties of Markov chains, and solving the equations to obtain the steady-state probability vector. The steady state probability vector of the Markov chain Z is given by $\Pi = [\pi_{(0,g)}, \pi_{(1,g)}, \ldots, \pi_{(l,g)}, \ldots, \pi_{(b_i,g)}]$ where every element $\pi_{(l,g)} = [\pi_{(l,g,1,1)}, \pi_{(l,g,1,2)}, \ldots, \pi_{(l,g,N,N)}], l > 0$ is a row vector of size N^2, except that $\pi_{(0,g)}$ is a scalar. For the steady state probabilities in level l, we denote it by $\pi_l = [\pi_{(l,0)}, \pi_{(l,1)}]$, where $\pi_{(l,0)}$ and $\pi_{(l,1)}$ are two probability vectors for the traffic source at input i in stage 0 and 1. Furthermore, we let $\overline{\pi_{(l,g)}} = \pi_{(l,g)}e_1$ and $\overline{\pi_0} = \pi_{(0,0)} + \pi_{(0,1)}$.

2.3 Solving the Markov Chain

We now derive the equations for computing the blocking probability, $P_{blo,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$ and the success probability, $P_{suc,W_t(w_i',w_o')|W_{t-1}(g,w_i,w_o)}$. The transition of the state of the virtual HOL input/output queues from the state (w_i, w_o) to state (w_i', w_o') is a two-step process as illustrated in Figure 2: (i) First, we account for the number (k_i, k_o) of the newly arriving HOL cells to the virtual HOL input/output queues; (ii) Then, we consider the transition from the intermediate state (h_i, h_o) to the final state (w_i', w_o') after applying the PIM algorithm.

Due to the space limit here, we omit the detailed

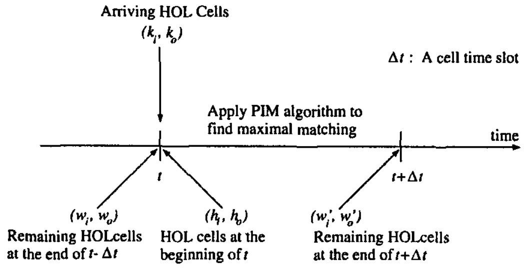


Figure 2: Transition of the virtual HOL queues.

procedure of deriving these transition probabilities. We will focus only on the basic idea of our solution for the computation of these probabilities. To utilize the concept of a *tagged queue*, the condition of independent and identical components must be satisfied. Studies indicate that such an assumption is reasonable for the moderate or large size input-queued switches under *i.i.d* traffics [6, 7] . Here we make the same assumption, that is, when a cell arrives at an empty queue $Q(i,j)$, it will automatically observe another jth queue being empty with Bernoulli probability p_0 and another queue in input i being empty with Bernoulli probability $\overline{\pi_0}$. The introduction of p_0 plays an essential role in the solution of the Markov chain Z . However, the difficulty is that p_0 can't be directly derived from the known system parameters, such as the switch size, buffer size and traffic load. Instead of assuming p_0 as a known parameter, we use the *fixed point* iterative method to obtain p_0 from the known system parameters [5]. Eq (1) gives the *fixed point equation.* In particular, we prove a lemma which states that a fixed point for Eq (1), given below, exists.

$$p_0 = (1 - \frac{1-\beta}{N})\pi_{(0,0)} + (1 - \frac{\alpha}{N})\pi_{(0,1)} \qquad (1)$$

Lemma 1 *A fixed point exists for Eq (1) in the interval [0,1].*

Proof: Let the measure derived from the Markov chain be the probability p_0. We know that both α and β are *constants.* According to the Rules (1) and (4) of THEOREM 2 in [4], a fixed point exists. In addition,

$$p_0 = (1-\frac{1-\beta}{N})\pi_{(0,0)} + (1-\frac{\alpha}{N})\pi_{(0,1)} \leq \pi_{(0,0)} + \pi_{(0,1)} \leq 1$$

Thus, the fixed point must exist in interval [0, 1]. ∎

Given p_0, the formula for the probability of the *virtual HOL queue*'s transition from (h_i, h_o) to (w_i, w_o) can be derived with some efforts [6, 7]. Consequently, the steady state probabilities of Z are given by:

$$\pi_1 C_0 + \pi_0 C_1 = \pi_0 \qquad (2)$$

$$\pi_0([1,1]^T + \sum_{i=1}^{b_i} \prod_{j=1}^{i} \alpha_j e) = 1 \qquad (3)$$

$$\pi_i = \pi_0 \prod_{j=1}^{i} \alpha_j, \ for\ 1 \leq i \leq b_i. \qquad (4)$$

where α_i is given as

$$\begin{cases} A_2(I - D_1)^{-1}, & for\ i = b_i; \\ A_2(I - A_1 - \alpha_{b_i} D_0)^{-1}, & for\ i = b_i - 1; \\ A_2(I - A_1 - \alpha_{i+1} A_0)^{-1}, & for\ i \in [2, b_i - 2]; \\ C_2(I - A_1 - \alpha_2 A_0)^{-1}, & for\ i = 1. \end{cases} \qquad (5)$$

The elements of matrices in Eq (5) are functions of $\pi_{(0,0)}$, $\pi_{(0,1)}$, $\overline{\pi_{(1,0)}}$ and $\overline{\pi_{(1,1)}}$. This naturally suggests an iterative solution [3, 6, 7]. Initially, both $\pi_{(0,0)}$ and $\pi_{(0,1)}$ are set to be $0.5(1-\lambda)$, which corresponds to the cases that there is no new arriving cell at the *tagged input queue* at the beginning of a time slot, and both $\overline{\pi_{(1,0)}}$ and $\overline{\pi_{(1,1)}}$ are approximated by $0.5(1-N^{-1})\lambda(\pi_{(0,0)} + \pi_{(0,1)})$. Then the next $\pi_{(0,0)}$ and $\pi_{(0,1)}$ are obtained by finding the root for Eq (2) and Eq (3). Consequently, the new π_1 is computed by Eq (4). As observed from our experiments, the converging rate is quite high and an accuracy of 10^{-5} for π_0 can be attained within 15 iterative computations in most of cases.

2.4 Computing the Performance Metrics

So far, we have solved the underlying Markov chain of our queueing model for the PIM switch. From the symmetry property of the model, some interesting performance parameters of other input queues, such as throughput ρ, mean queue length $\overline{Q}$, mean cell delay $\overline{D}$ and mean cell loss probability P_{loss} are the same as which given below for the *tagged input queue*:

$$\rho = [\pi_{(0,0)}(\lambda_0 - B_0^{(0)} e_1) + \pi_{(0,1)}(\lambda_1 - B_0^{(1)} e_1)] + \sum_{l=1}^{b_i} [\pi_{(l,0)}(S^{(0)} + S'^{(0)}) + \pi_{(l,1)}(S^{(1)} + S'^{(1)})] e_1$$

$$\bar{Q} = \sum_{l=1}^{b_i} l \pi_l e \qquad \bar{D} = \bar{Q}/\rho$$

$$P_{loss} = (\pi_{(b_i,0)}\lambda_0 + \pi_{(b_i,1)}\lambda_1) e_1/\lambda$$

3 Numerical Results

Both mathematical analysis and simulation results are presented in this section in order to investigate the accuracy of the above queueing model and to evaluate the performance of the PIM switch under bursty traffic. Figure 3.(a), (b) and (c) show the switch throughput, mean cell delay and mean cell loss probability as function of offered load with a mean burst length of 8 cells for an 8×8 PIM switch with various PIM scheduling iteration numbers 1, 2 and 3, respectively. For the mean cell loss probability, simulation results are given only in case of the switches being overloaded under given system configurations so that the results obtained by simulation are reasonable. It can be seen from these figures that the mathematical analysis results closely approximate the simulation results. Noticeable deviations between the analysis and simulation appear only in cases where the switch with multiple iterations is *overloaded.*

4 Conclusion

The presented analysis provides a unifying framework to build queueing models for PIM switches. In addition, the queueing model can be extended using the same technique to the situation where complicated bursty traffics with more states are asserted to the switch. Recalling our previous work in [6, 7], we conclude that our suggested queueing model works well not only in case of the *i.i.d* Bernoulli traffic, but also in case of the *i.i.d* burst traffic where the cells' arrival process is correlated in a long term.

References

[1] A. Adas. Traffic models in broadband networks. *IEEE Communications Magazine*, 35(7):82–89, July 1997.

[2] T. E. Anderson, S. S. Owicki, J. B. Saxe, and C. P. Thacker. High-speed switch scheduling for local-area networks. *ACM Transactions on Computer Systems*, 11(4):319–352, Nov. 1993.

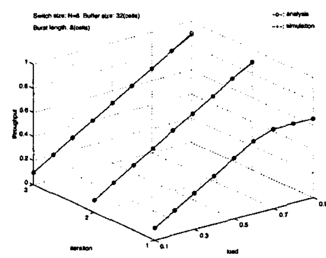
[3] Y. C. Jung and C. K. Un. Performance analysis of packet switches with input and output buffers. *Computer Networks and ISDN Systems*, 26(12):1559–1580, Sept. 1994.

[4] V. Mainkar and K. S. Trivedi. Sufficient conditions for existence of a fixed point in stochastic reward net-based iterative models. *IEEE Transactions on Software Engineering*, 22(9):640–53, Sept. 1996.

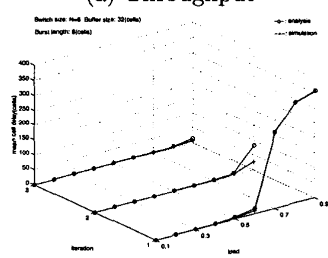
[5] M. K. Mehmet-Ali, M. Youssefi, and H. T. Nguyen. The performance analysis and implementation of an input access scheme in a high-speed packet switch. *IEEE Trans. on Communications*, 42(12):3189–3199, Dec. 1994.

[6] G. Nong, J. K. Muppala, and M. Hamdi. A performance model for ATM switches with multiple input queues. *Proceedings IC3N'97, pp. 222-227, Las Vegas*, Sep. 1997.

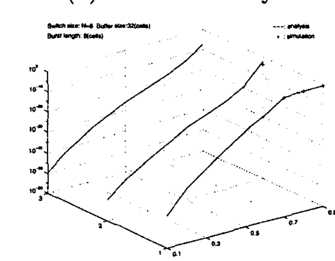
[7] G. Nong, J. K. Muppala, and M. Hamdi. Analysis of non-blocking ATM switches with multiple input queues. *IEEE/ACM transactions on networking*, Feb. 1999.



(a) Throughput



(b) Mean cell delay



(c) Mean cell loss probability

Figure 3: The throughput, mean cell delay and mean cell loss probability of an 8-by-8 PIM switch with buffer sizes $b_i = 32$, as a function of offered loads with mean burst lengths $\tau = 8$.