ELSEVIER

Computer Networks and ISDN Systems 28 (1996) 1809–1829

COMPUTER NETWORKS and ISDN SYSTEMS

Efficient analysis of Multistage Interconnection Networks using finite output-buffered switching elements [1]

Bin Zhou [a], M. Atiquzzaman [b,*]

[a] *Motorola Australia Software Centre, 2 Second Ave., Technology Park, Adelaide, SA 5095, Australia*

[b] *Department of Electrical and Computer Systems Engineering, Monash University, Clayton 3168, Melbourne, Australia*

Abstract

The performance of Multistage Interconnection Networks (MINs) constructed from output buffered switching elements (SE) is higher than those having input buffered SEs. Many of the existing analytical models for output buffered MINs assume uniform traffic and infinite buffers at each output port of an SE. The models are not realistic because, in practice buffers are finite and the traffic may not be uniform. Moreover, because of simplifying assumptions, the models do not produce *accurate* results. For the purpose of network design and proper buffer dimensioning, it is important to develop an accurate analytical model under realistic traffic patterns and finite buffered SEs. The objective of this paper is to develop an *accurate* model for MINs using *finite* output buffered SEs and operating in the presence of *nonuniform traffic* patterns. It is shown that the proposed analytical model is much *more* accurate than existing models.

Keywords: Broadband ISDN; ATM networks; ATM switching; Performance evaluation; Interconnection networks; Performance modelling

1. Introduction

Multistage interconnection networks (MINs) are used to connect processors and memories in large-scale scalable multiprocessor systems. They have also been proposed as ATM switching nodes in Broadband ISDN networks. Many architectures have been studied [1–7] in an effort to realize a high-speed ATM switch with low packet loss probability. A single stage network is characterized by a stage of switching elements which are connected to the inputs and outputs of the network by a specific connection pattern. A well-known representative of these networks is the *shuffle exchange* network [8], which is based on a perfect shuffle permutation cascaded to a stage of switching elements. This network is also called a *recirculating* network, because it may be necessary to route a packet through the network several times before it can reach the proper destination. Since the performance of single stage networks is low, usually several stages of SEs are cascaded forming the so-called multistage networks. Multistage networks are built of several stages which are

* Corresponding author. Email: atiq@eng.monash.edu.au.

[1] A part of this paper has been presented at the First IEEE International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'95), April 19–21, 1995, Brisbane, Australia. The work was carried out when the authors were with the Computer Science & Computer Engineering Department at LaTrobe University, Bundoora, Australia.

PII S0169-7552(96)00003-7

Table 1
Summary of different buffering strategies for $a \times a$ SEs

Buffer strategy	Structure	Internal speed	Throughput	Special
input	simple	equal to input	limited	head of line
output	complex	a times of input	achieve optimal	separate buffer
crosspoint	simple	equal to input	better	reduce HOL
shared	complex	a times of input	very high	buffer hogging

Table 2
Summary of existing analytical models

Buffer strategy	Uniform	General	Hot spot	SSSD	Bursty
input	[1,12–15]	[16]	[17]	[16]	[18]
output	[2,19]	[20–22]	[21,22]	[23]	[18]
crosspoint	[24,7]	[25]	[25]		
shared	[5]			[26]	[6,4]

interconnected by certain link patterns. Unlike single stage networks, no recirculation of packets is necessary resulting in a higher throughput.

A MIN (see Fig. 3) consists of several stages of small crossbar switching elements (SE). The successive stages are connected by a permutation function. Many of the MINs have buffers to store blocked packets. The buffering schemes in MINs include the input buffer, output buffer, shared buffer and crosspoint buffer [9,10]. Each scheme has its own advantages and disadvantages. Some schemes perform better under a uniform input traffic pattern while others are better under a non-uniform traffic pattern like the hot spot and single-source-to-single-destination (SSSD) traffic patterns. For example, shared buffer has a high buffer utilization and throughput under uniform traffic pattern although it suffers from a phenomenon called *buffer hogging* [11] in the case of a non-uniform traffic pattern. Table 1 summarizes the characteristics of the different buffering schemes. Work carried out on the performance of the different buffering schemes under uniform, general, hot spot, single-source-to-single-destination and bursty traffic patterns are consolidated in Table 2.

The overall performance of a multiprocessor system or the B-ISDN network depends heavily on the performance of the interconnection network or the switching node respectively. Because of the excessive computing power and time required to study the performance of networks using simulations, analytical models are usually developed to facilitate such study. Early work in the performance analysis of unbuffered MINs was done by Patel [27], who derived a recursive formula for the evaluation of the throughput of unbuffered MINs. The analysis of MINs constructed of single input-buffered SEs was pioneered by Jenq [1]. Subsequently, the model of Jenq was extended by various authors to model networks constructed from switching elements of arbitrary size, more complex SEs with buffers placed at the outputs or more general traffic patterns [2,12,16]. Theimer proposed a refined model to analyze the performance of single-buffered MINs with 2×2 SEs and demonstrated a significant improvement in accuracy [13]. Mun and Youn [15] investigated in detail the accuracy of the models in [1,12] and pointed out why those models were not accurate enough. Then they presented a new model, which demonstrated a high degree of accuracy for input buffered MINs. It is to be mentioned that most of the above models have assumed MINs with input-buffered SEs.

It is well known that due to the head-of-line blocking, the throughput of input-buffered MINs is lower than those using output-buffered SEs. Several analytical models have been proposed for the performance evaluation of output buffered MINs [2,21]. Kruskal [28] has analyzed infinite-buffered MINs and Kim [2] and James [19] have studied finite output-buffered SEs respectively. The models are either not very accurate or assume traffic which is uniformly distributed over the outputs. Uniform traffic does not necessarily represent the traffic in a real system. Several non-uniform traffic patterns, viz. hot spot [29], favorite output [30] and single-source-to-single-destination [16] have been reported in the literature. Most of the existing models do not produce accurate results under non-uniform traffic, especially when the input traffic load is high. The inaccuracies are

mainly due to simplifying assumptions regarding blocked packets in the network. Since non-uniform traffic results in a significant blocking in the network, the results from the existing models are not very accurate.

In this paper, a new accurate model for evaluating the performance of MINs using finite output-buffered 2×2 SEs is proposed. Previous models [1,13] have assumed a renewal process for blocked packets. They assume that a blocked packet can choose a new destination in a subsequent cycle. The proposed model *differs* from existing models because it accounts for the fact that a packet which is blocked in a buffer always hunts for the same output link in subsequent clock cycles. It takes a rigorous account of this behavior of blocked packets and also considers the dependency between packets in consecutive clock cycles and states of the buffers in adjacent stages. Due to taking into consideration the above facts, it *produces results which are more accurate* than existing models. Moreover, it is *applicable to both uniform and non-uniform traffic* patterns in the network. The accurate model presented in this paper will serve as a *tool* for the network designer, and will help in obtaining a *better insight* into the performance of the network under different non-uniform traffic patterns. The *objectives* of this paper are as follows:

- To develop an accurate analytical model which can be applied to evaluate the performance of MINs constructed of 2×2 SEs having arbitrary sized buffers at the outputs and operating under a general input traffic pattern.
- To model the queueing of packets at the input of the network and resubmit them in subsequent clock cycles.
- To take into consideration in the model, the routing behavior of blocked packets in consecutive cycles.
- To compare the results obtained from existing model, the proposed model and simulation for both uniform and nonuniform traffic patterns.

In Section 2, the structure, operation, and modeling assumptions of an output-buffered Omega network are described. The proposed model for the performance evaluation of an output-buffered Omega network under a general traffic environment is proposed in Section 3. The uniform and non-uniform traffic patterns that are used to test the robustness of the model are described in Section 4. Results obtained from the proposed model and previous models are compared with those from simulations in Section 5, followed by concluding remarks in Section 6.

2. Model platform

The Omega network [27] will be taken as an example of a MIN to be modeled. The model, in fact, applies equally well to all unique path networks. We make the following assumptions regarding the operation and environment of the interconnection network as in [2,17,21]:

(1) There are $N = 2^s$ inputs and N outputs in the network. Each input of the network has an Input Buffer Controller (IBC) of finite size.

(2) The network operates *synchronously*. This implies that packets move from one stage to the next only at the beginning of a clock cycle. This reflects the situation in an ATM environment where all packets (called cells) have a fixed length, and fit exactly into one clock cycle. For modeling purposes, we split the clock cycle into two phases [13].
   - In the first phase, the availability of buffer space at the succeeding stage along the destined path of a packet is determined.
   - Depending on the availability of buffer space in the succeeding stage, a packet may move forward one stage in the second phase.

(3) A *backpressure* mechanism ensures that no packets are lost within the network.

(4) Packet arrival at the inputs of the network are *Bernoulli* processes and are independent of each other.

(5) Each input link of the network is offered the same traffic *load*.

(6) There is no *blocking* at the output links of the network.

(7) The *conflict resolution* logic at each SE is unbiased.
(8) A packet must spend at least one clock cycle in each buffer including the IBC.
(9) If there are more than two spaces available at the destination buffer in the next stage, the SE is assumed to be fast enough to accept both packets in one cycle.

3. The model

We develop a Markov chain model for finite-buffered MINs in this section. The analysis is based on the methods of [13,16,17,21]. Fig. 1 shows three SEs in consecutive stages of the network. The input and output ports of the lth SE at stage k are denoted by klx, $kl\underline{x}$, kly and $kl\underline{y}$. Let $(k+1)\hat{l}\hat{y}$ denote the input port of the SE at the $(k+1)$th stage, which is fed by the output port kly. Also, let $(k-1)\tilde{l}\tilde{x}$ denote the output port of the SE at the $(k-1)$th stage, which feeds the input port klx.

In defining the state variables, a *new* state means that the packet at the head of the queue has moved to the head of the queue in the previous network cycle. A *blocked* state implies that the packet at the head of the queue has made at least one unsuccessful routing attempt. The state variables for a buffer at an SE are as follows:

- State 0: The buffer is empty.
- State (i,n): The buffer contains i, $1 \le i \le m$, packets and the packet at the head of the queue is in a new state.
- State (i,b): The buffer contains i, $1 \le i \le m$, packets and the packet at the head of queue is in a blocked state.

From the definitions, it is clear that when a buffer is not empty, it will be in either state (i,n) or (i,b). When a buffer is in state (i,b), the packet at the head of queue either lost the contention with another packet competing for the same output link, or the receiving buffer at the succeeding stage was full. A buffer in state (i,n) implies that the destination buffer could be in state 0, new or blocked. The probability that a packet can move forward during a cycle will strongly depend on the situation during the previous cycle. Models which consider this state dependency between consecutive clock cycles will have higher accuracy.

The states of the buffer at the beginning of phase 1, at the end of phase 1, and at the end of phase 2 of a stage cycle will be called the *initial* state, the *intermediate* state, and the *final* state respectively. The initial state $B^i_{klv}(\beta,t)$, intermediate state $\tilde{B}^i_{klv}(\beta,t)$ and other probabilities of a buffer at port klv, $v \in \{y,\underline{y}\}$, during cycle t are defined as follows:

$B^0_{klv}(0,t)$ = probability that a buffer at klv, $v \in \{y,\underline{y}\}$, is in empty state at the end of cycle t.
$B^i_{klv}(\beta,t)$ = probability that a buffer at klv, $v \in \{y,\underline{y}\}$, containing i, $0 \le i \le m$, packets is in state β, $\beta \in \{n,b\}$, at the end of cycle t.
$\tilde{B}^0_{klv}(0,t)$ = probability that a buffer at klv, $v \in \{y,\underline{y}\}$, is in empty state at the end of phase 1 of cycle t.
$\tilde{B}^i_{klv}(\beta,t)$ = probability that a buffer at klv, $v \in \{y,\underline{y}\}$, containing i, $0 \le i \le m$, packets is in state β, $\beta \in \{n,b\}$, at the end of phase 1 of cycle t.
λ = probability that a packet is ready to come to an IBC buffer at the beginning of a cycle.
s = number of switching stages in the network.
m, f = size of buffers at the output of an SE and at the IBC, respectively.
P_{kluv} = probability that a new packet, ready to come to klu, $u \in \{x,\underline{x}\}$, is destined to klv, $v \in \{y,\underline{y}\}$.
$Q_{klu}(*,t)$ = probability that a packet (new and/or blocked) is ready to come to klu, $u \in \{x,\underline{x}\}$, during cycle t.
$Q_{klu}(n,t)$ = probability that a new packet is ready to come to port klu, $u \in \{x,\underline{x}\}$, during cycle t.
$Q_{klu}(b,t)$ = probability that a blocked packet is ready to come to port klu, $u \in \{x,\underline{x}\}$, during cycle t.
$C^i_{klv}(n,t)$ = probability that i , $0 \le i \le 2$, new packets are ready to come to the buffer at klv, $v \in \{y,\underline{y}\}$, during cycle t.

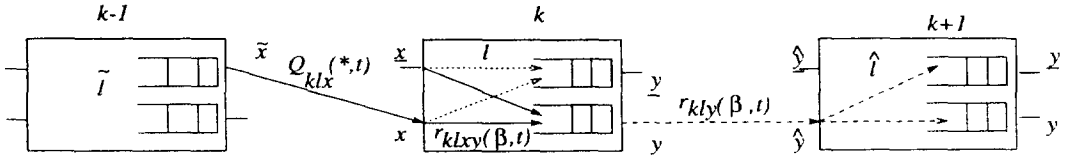


Fig. 1. The illustration of $Q_{klx}(*,t)$, $r_{klxy}(\beta,t)$, and $r_{kly}(\beta,t)$.

$C^i_{klv}(*,t)$ = probability that i, $0 \le i \le 2$, packets (new and/or blocked) are ready to come to the buffer at klv, $v \in \{y, \underline{y}\}$, during cycle t.

The routing probability of a packet is defined as the probability that a packet in the buffer of an SE can be routed to the next stage. The routing and blocking probabilities will be denoted by r and $\bar{r}$ respectively and are defined as follows:

$r_{kluv}(\beta,t)$ = probability that a packet from port klu, $u \in \{x, \underline{x}\}$, advances to the buffer klv, $v \in \{y, \underline{y}\}$, during phase 2 of clock cycle t, given that a packet in state β, $\beta \in \{n, b\}$, destined to klv, is ready to come to klu during cycle t.

$r_{klv}(\beta,t)$ = probability that a packet at the head of the queue in the buffer at klv, $v \in \{y, \underline{y}\}$, is in state β, $\beta \in \{n, b\}$, and is able to move forward to the next stage during phase 1 of cycle t.

We define $\bar{z} = 1 - z$ for any variable z. Now, we obtain equations for $Q_{klu}(*,t)$, $r_{kluv}(\beta,t)$, and $r_{klv}(\beta,t)$. Their relationship is shown in Fig. 1. A packet, ready to come to port klu, $u \in \{x, \underline{x}\}$, and destined to port klv, $v \in \{y, \underline{y}\}$, can pass from klu to the buffer at klv when the following conditions are satisfied:

(1) There are at least two packet spaces in the buffer at port klv, or
(2) the buffer at klv has $m-1$ packets and a packet (new or blocked) in the buffer advances to the next stage in the same clock cycle, or
(3) the buffer at klv has $m-1$ packets and no packet can be forwarded, and there is no conflict between the packet (new or blocked) at klu and a possible packet at $kl\underline{u}$, or there is a packet (new or blocked) at $kl\underline{u}$ which is also destined to klv and the conflict is resolved in favor of the packet at klu, or
(4) the buffer at klv has m packets and a packet (new or blocked) advances to the next stage in the same cycle, and there is no conflict between the packet at klu and a possible packet at $kl\underline{u}$, or there is a packet at $kl\underline{u}$ which is destined to klv, and the conflict is resolved in favor of the packet at klu.

The above conditions translate into the following equation for the probability that a packet in new state, ready to come to the klu port and destined to klv, can advance to the buffer at klv:

$$\begin{aligned} r_{kluv}(n,t) = 1 &- \{B^m_{klv}(n,t-1) + B^m_{klv}(b,t-1) + B^{m-1}_{klv}(n,t-1) + B^{m-1}_{klv}(b,t-1)\} \\ &+ B^{m-1}_{klv}(n,t-1) r_{klv}(n,t) + B^{m-1}_{klv}(b,t-1) r_{klv}(b,t) \\ &+ \{B^{m-1}_{klv}(n,t-1)\bar{r}_{klv}(n,t) + B^{m-1}_{klv}(b,t-1)\bar{r}_{klv}(b,t) + B^m_{klv}(n,t-1) r_{klv}(n,t) \\ &+ B^m_{klv}(b,t-1) r_{klv}(b,t)\} \times \{1 - Q_{kl\underline{u}}(*,t) + (Q_{kl\underline{u}}(b,t) + Q_{kl\underline{u}}(n,t)) P_{kl\underline{uv}} \\ &+ 0.5(Q_{kl\underline{u}}(b,t) + Q_{kl\underline{u}}(n,t)) P_{kl\underline{uv}}\}. \end{aligned} \tag{1}$$

The probability that a packet at klv, $v \in \{y, \underline{y}\}$, can move forward is given by

$$r_{klv}(n,t) = P_{(k+1)\hat{l}\hat{v}y} r_{(k+1)\hat{l}\hat{v}y}(n,t) + P_{(k+1)\hat{l}\hat{v}\underline{y}} r_{(k+1)\hat{l}\hat{v}\underline{y}}(n,t). \tag{2}$$

The packet arrival probability $Q_{klu}(*,t)$ can be obtained from the sum of the probabilities that the upstream buffer at stage $k-1$ is not empty.

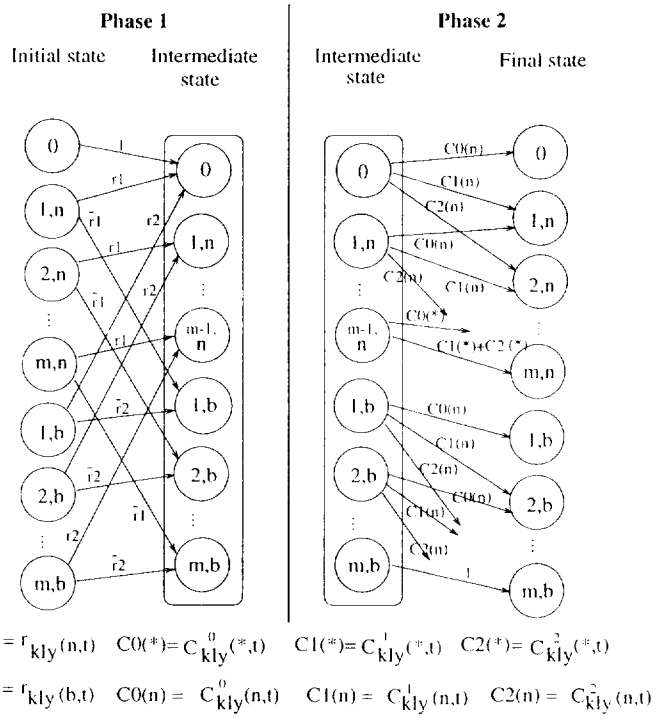


Fig. 2. Markov state transition diagram for internal stage buffers.

$$Q_{klu}(n,t) = \sum_{i=1}^{m} B^{i}_{(k-1)\tilde{l}\tilde{u}}(n,t-1), \tag{3}$$

$$Q_{klu}(b,t) = \sum_{i=1}^{m} B^{i}_{(k-1)\tilde{l}\tilde{u}}(b,t-1), \tag{4}$$

$$Q_{klu}(*,t) = 1 - B^{0}_{(k-1)\tilde{l}\tilde{u}}(0,t-1). \tag{5}$$

When a packet in stage $k-1$ is in state (i,b) at the end of cycle t, the destination buffer in stage k after phase 1 of cycle $t+1$ will be either full or will have one space available. If a packet was blocked at stage $k-1$ at the end of cycle $t-1$, the destination buffer of this packet at stage k must be full at the end of cycle $t-1$. Under this condition,

- the destination buffer in stage k at the end of cycle $t-1$ can be in state (m,n) only if a packet left the buffer during phase 1 of cycle $t-1$ and a packet arrived from another buffer in stage $k-1$ during cycle $t-1$, or
- the destination buffer in stage k at the end of cycle $t-1$ can be in state (m,b) under one of the following conditions:
  (1) Buffer was full at the end of cycle $t-2$ and no packet left the buffer during cycle $t-1$.
  (2) Buffer had one space at the end of cycle $t-2$, no packet left during cycle $t-1$, and one packet came during cycle $t-1$.

Under the above conditions, the intermediate state of the destination buffer in stage k during cycle t will be either full if it is blocked or one space available if a packet left during cycle t. We define $B'^{m}_{klv}(\beta,t)$, $\beta \in \{n,b\}$, as the probability of this buffer being full and in state β at the end of cycle t, given that a packet at stage $k-1$

which is blocked for klv is ready to come to klu at the end of cycle t. From the above justification, we can obtain $B'^{m}_{klv}(\beta,t)$ as follows:

$$B'^{m}_{klv}(n,t-1) = \{B^{m}_{klv}(n,t-2)r_{klv}(n,t-1) + B^{m}_{klv}(b,t-2)r_{klv}(b,t-1)\} \times \{0.5(Q_{kl\underline{u}}(n,t-1) + Q_{kl\underline{u}}(b,t-1))P_{kl\underline{uv}}\}, \quad (6)$$

$$B'^{m}_{klv}(b,t-1) = 1 - B'^{m}_{klv}(n,t-1). \quad (7)$$

The blocked packet in stage $k-1$ can be routed during cycle t if the packet wins any possible contention from the other buffer in stage $k-1$ which feeds the same destination buffer and destination buffer has one space after phase 1 of cycle t.

$$r_{kluv}(b,t) = \{1 - Q_{kl\underline{u}}(*,t) + (Q_{kl\underline{u}}(b,t) + Q_{kl\underline{u}}(n,t))P_{kl\underline{uv}} + 0.5(Q_{kl\underline{u}}(b,t) + Q_{kl\underline{u}}(n,t))P_{kl\underline{uv}}\} \times \{B'^{m}_{klv}(n,t-1)r_{klv}(n,t) + B'^{m}_{klv}(b,t-1)r_{klv}(b,t)\}, \quad (8)$$

$$r_{klv}(b,t) = P_{(k+1)\hat{l}\hat{v}y}r_{(k+1)\hat{l}\hat{v}y}(b,t) + P_{(k+1)\hat{l}\hat{v}\underline{y}}r_{(k+1)\hat{l}\hat{v}\underline{y}}(b,t). \quad (9)$$

The number of packets ready to come to buffer klv are given by

$$C^{0}_{klv}(n,t) = \{1 - Q_{kl\underline{u}}(n,t) + Q_{kl\underline{u}}(n,t)P_{kl\underline{uv}}\}\{1 - Q_{klu}(n,t) + Q_{klu}(n,t)P_{klu\underline{v}}\}, \quad (10)$$

$$C^{1}_{klv}(n,t) = Q_{klu}(n,t)P_{kluv}\{1 - Q_{kl\underline{u}}(n,t) + Q_{kl\underline{u}}(n,t)P_{kl\underline{uv}}\} + Q_{kl\underline{u}}(n,t)P_{kl\underline{uv}}\{1 - Q_{klu}(n,t) + Q_{klu}(n,t)P_{klu\underline{v}}\}, \quad (11)$$

$$C^{2}_{klv}(n,t) = Q_{klu}(n,t)P_{kluv}Q_{kl\underline{u}}(n,t)P_{kl\underline{uv}}, \quad (12)$$

$$C^{0}_{klv}(*,t) = \{1 - Q_{kl\underline{u}}(*,t) + (Q_{kl\underline{u}}(b,t) + Q_{kl\underline{u}}(n,t))P_{kl\underline{uv}}\} \times \{1 - Q_{klu}(*,t) + (Q_{klu}(b,t) + Q_{klu}(n,t))P_{klu\underline{v}}\}, \quad (13)$$

$$C^{1}_{klv}(*,t) = \{Q_{klu}(n,t) + Q_{klu}(b,t)\}P_{kluv}\{1 - Q_{kl\underline{u}}(*,t) + (Q_{kl\underline{u}}(b,t) + Q_{kl\underline{u}}(n,t))P_{kl\underline{uv}}\} + \{Q_{kl\underline{u}}(n,t) + Q_{kl\underline{u}}(b,t)\}P_{kl\underline{uv}}\{1 - Q_{klu}(*,t) + (Q_{klu}(b,t) + Q_{klu}(n,t))P_{klu\underline{v}}\}, \quad (14)$$

$$C^{2}_{klv}(*,t) = \{Q_{klu}(n,t) + Q_{klu}(b,t)\}P_{kluv}\{Q_{kl\underline{u}}(n,t) + Q_{kl\underline{u}}(b,t)\}P_{kl\underline{uv}}. \quad (15)$$

3.1. Internal stage buffers

Fig. 2 shows the $2m+1$ state Markov chain state transitions from the initial states of a buffer to intermediate states and from intermediate states to final states, for a buffer in the internal stages of the MIN, i.e. the stages other than the first and the last one. Some of the state transitions in Fig. 2 are explained as follows.

A buffer at initial state (i,n) may go to intermediate state $(i-1,n)$ if the destined buffer at the next stage allows a packet at this stage to move forward, and the packet can win any possible conflict with another packet in the same stage or may go to state (i,b) if the destined buffer at the next stage does not accept a packet or the packet loses any possible conflict. A buffer at initial state (i,b) may go to intermediate state $(i-1,n)$ if the destined buffer at the next stage allows a packet at this stage to move forward and the packet can win any possible conflict with another packet in the same stage, or may stay at the state (i,b) if the destined buffer at the next stage does not allow a packet at this stage to move forward or the packet loses any possible conflict. A buffer may go from intermediate state 0 to final state 0, $(1,n)$ or $(2,n)$ if none, one or two packets are offered to the buffer respectively. A buffer from intermediate state $(m-1,n)$ may go to final state (m,n) if one or two packets are offered to the buffer, and stay in the final state $(m-1,n)$ if no packet is offered. A buffer may go from intermediate state $(m-1,b)$ to final state (m,b) if one or two packets are offered to the

buffer, or stay in state $(m-1,b)$ if no packet is offered. A buffer at intermediate state (m,b) will stay in the same state in all cases.

We introduce the final state matrix $\boldsymbol{B}_{klv}(*,t)$, intermediate state matrix $\tilde{\boldsymbol{B}}_{klv}(*,t)$, and routing matrix $\boldsymbol{r}_{klv}(*,t)$ as follows:

$$\boldsymbol{B}_{klv}(*,t)=\begin{bmatrix} B^0_{klv}(0,t) & 0 & \dots & 0 \\ B^1_{klv}(n,t) & B^2_{klv}(n,t) & \dots & B^m_{klv}(n,t) \\ B^1_{klv}(b,t) & B^2_{klv}(b,t) & \dots & B^m_{klv}(b,t) \end{bmatrix},$$

$$\tilde{\boldsymbol{B}}_{klv}(*,t)=\begin{bmatrix} \tilde{B}^0_{klv}(0,t) & \tilde{B}^1_{klv}(n,t) & \dots & \tilde{B}^{m-1}_{klv}(n,t) \\ \tilde{B}^1_{klv}(b,t) & \tilde{B}^2_{klv}(b,t) & \dots & \tilde{B}^m_{klv}(b,t) \end{bmatrix},$$

$$\boldsymbol{r}_{klv}(*,t)=\begin{bmatrix} 1 & r_{klv}(n,t) & r_{klv}(b,t) \\ 0 & \bar{r}_{klv}(n,t) & \bar{r}_{klv}(b,t) \end{bmatrix}.$$

Using Fig. 2, the intermediate state probabilities are calculated as follows:

$$\tilde{\boldsymbol{B}}_{klv}(*,t)=\boldsymbol{r}_{klv}(*,t)\boldsymbol{B}_{klv}(*,t-1). \tag{16}$$

Since at most one packet can leave a buffer during a cycle, the intermediate state of the buffer at klv during cycle t can not be $(m-2)$ if the buffer was full at the end of cycle $(t-1)$. Therefore, only new packets can come to buffer klv if the intermediate state of the buffer is $(m-2)$ or less.

To formulate the expressions for the state transitions illustrated in Fig. 2, we introduce the two-dimensional matrices $\boldsymbol{C''}_{klv}$ and $\boldsymbol{C'}_{klv}$ of sizes $m\times(m+1)$ and $m\times m$ respectively for $1\le k\le s$.

$$\boldsymbol{C''}_{klv}=\begin{bmatrix} C^0_{klv}(n) & C^1_{klv}(n) & C^2_{klv}(n) & 0 & \dots & 0 & 0 \\ 0 & C^0_{klv}(n) & C^1_{klv}(n) & C^2_{klv}(n) & \dots & 0 & 0 \\ 0 & 0 & C^0_{klv}(n) & C^1_{klv}(n) & \dots & 0 & 0 \\ 0 & 0 & 0 & C^0_{klv}(n) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & C^2_{klv}(n) & 0 \\ 0 & 0 & 0 & 0 & \dots & C^1_{klv}(n) & C^2_{klv}(n) \\ 0 & 0 & 0 & 0 & \dots & C^0_{klv}(*) & C^2_{klv}(*)+C^1_{klv}(*) \end{bmatrix},$$

$$\boldsymbol{C'}_{klv}=\begin{bmatrix} C^0_{klv}(n) & C^1_{klv}(n) & C^2_{klv}(n) & 0 & \dots & 0 & 0 \\ 0 & C^0_{klv}(n) & C^1_{klv}(n) & C^2_{klv}(n) & \dots & 0 & 0 \\ 0 & 0 & C^0_{klv}(n) & C^1_{klv}(n) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & C^1_{klv}(n) & C^2_{klv}(n) \\ 0 & 0 & 0 & 0 & \dots & C^0_{klv}(*) & C^1_{klv}(*)+C^2_{klv}(*) \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}.$$

We also define $\boldsymbol{B}_{klv}(n,t)$, $\boldsymbol{B}_{klv}(b,t)$, $\tilde{\boldsymbol{B}}_{klv}(n,t)$ and $\tilde{\boldsymbol{B}}_{klv}(b,t)$ as follows:

$$\boldsymbol{B}_{klv}(n,t)=\begin{bmatrix} B^0_{klv}(0,t) & B^1_{klv}(n,t) & \dots & B^m_{klv}(n,t) \end{bmatrix},\quad 0\le k\le s,$$

$$\boldsymbol{B}_{klv}(b,t) = \left[\begin{array}{cccc} B^1_{klv}(b,t) & B^2_{klv}(b,t) & \ldots & B^m_{klv}(b,t) \end{array}\right], \quad 0 \le k \le s,$$

$$\tilde{\boldsymbol{B}}_{klv}(n,t) = \left[\begin{array}{cccc} \tilde{B}^0_{klv}(0,t) & \tilde{B}^1_{klv}(n,t) & \ldots & \tilde{B}^{m-1}_{klv}(n,t) \end{array}\right], \quad 0 \le k \le s,$$

$$\tilde{\boldsymbol{B}}_{klv}(b,t) = \left[\begin{array}{cccc} \tilde{B}^1_{klv}(b,t) & \tilde{B}^2_{klv}(b,t) & \ldots & \tilde{B}^m_{klv}(b,t) \end{array}\right], \quad 0 \le k \le s.$$

Using Fig. 2, the state probabilities at the end of phase 2 are calculated from $\boldsymbol{C}'_{klv}$ and $\boldsymbol{C}''_{klv}$ as follows:

$$\boldsymbol{B}_{klv}(n,t) = \tilde{\boldsymbol{B}}_{klv}(n,t)\boldsymbol{C}''_{klv}, \tag{17}$$

$$\boldsymbol{B}_{klv}(b,t) = \tilde{\boldsymbol{B}}_{klv}(b,t)\boldsymbol{C}'_{klv}. \tag{18}$$

For example, the equations (obtained from Eqs. (16), (17) and (18)) for a MIN having two buffers (m=2) at each output of an SE are as follows:

Intermediate state probabilities.

$$\tilde{B}^0_{klv}(0,t) = B^0_{klv}(0,t-1) + B^1_{klv}(n,t-1)r_{klv}(n,t) + B^1_{klv}(b,t-1)r_{klv}(b,t), \tag{19}$$

$$\tilde{B}^1_{klv}(n,t) = B^2_{klv}(n,t-1)r_{klv}(n,t) + B^2_{klv}(b,t-1)r_{klv}(b,t), \tag{20}$$

$$\tilde{B}^1_{klv}(b,t) = B^1_{klv}(n,t-1)\bar{r}_{klv}(n,t) + B^1_{klv}(b,t-1)\bar{r}_{klv}(b,t), \tag{21}$$

$$\tilde{B}^2_{klv}(b,t) = B^2_{klv}(n,t-1)\bar{r}_{klv}(n,t) + B^2_{klv}(b,t-1)\bar{r}_{klv}(b,t), \tag{22}$$

$$\tilde{B}^0_{klv}(0,t) + \tilde{B}^1_{klv}(n,t) + \tilde{B}^1_{klv}(b,t) + \tilde{B}^2_{klv}(b,t) = 1. \tag{23}$$

Final state probabilities. If the buffer at klv at the end of time t is not full, then there can be no blocked packet in any of its source buffers in stage $(k-1)$. Moreover, since at most one packet can leave a buffer during a cycle, the intermediate state of the buffer at klv during cycle t cannot be zero if the buffer was full at the end of cycle $(t-1)$. Therefore, only new packets can come to buffer klv if the intermediate state of the buffer is zero.

$$B^0_{klv}(0,t) = \tilde{B}^0_{klv}(0,t)C^0_{klv}(n,t), \tag{24}$$

$$B^1_{klv}(n,t) = \tilde{B}^0_{klv}(0,t)C^1_{klv}(n,t) + \tilde{B}^1_{klv}(n,t)C^0_{klv}(*,t), \tag{25}$$

$$B^2_{klv}(n,t) = \tilde{B}^0_{klv}(0,t)C^2_{klv}(n,t) + \tilde{B}^1_{klv}(n,t)\{C^1_{klv}(*,t) + C^2_{klv}(*,t)\}, \tag{26}$$

$$B^1_{klv}(b,t) = \tilde{B}^1_{klv}(b,t)C^0_{klv}(*,t), \tag{27}$$

$$B^2_{klv}(b,t) = \tilde{B}^1_{klv}(b,t)\{C^1_{klv}(*,t) + C^2_{klv}(*,t)\} + \tilde{B}^2_{klv}(b,t), \tag{28}$$

$$B^0_{klv}(0,t) + B^1_{klv}(n,t) + B^2_{klv}(n,t) + B^1_{klv}(b,t) + B^2_{klv}(b,t) = 1. \tag{29}$$

3.2. *Last stage buffers*

The IBC buffer and the last stage need to be separately considered because they have different conditions from the internal stage buffers. The Markov chain at the last stage remains the same as that of the internal stages except that $r_{slv}(*,t) = 1$. Hence, we obtain the following equations:

$$\tilde{\boldsymbol{B}}_{slv}(*,t) = \boldsymbol{B}_{slv}(*,t-1), \tag{30}$$

$$\boldsymbol{B}_{slv}(n,t) = \tilde{\boldsymbol{B}}_{slv}(n,t)\boldsymbol{C}''_{slv}, \tag{31}$$

$$\boldsymbol{B}_{slv}(b,t) = [0\,0\,0\ldots0], \tag{32}$$

where

$$r_{slv}(*,t) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

For example, the equations for the last stage of a MIN having two buffers per SE are as follows.

Intermediate state probabilities.

$$\tilde{B}^0_{slv}(0,t) = B^0_{slv}(0,t-1) + B^1_{slv}(n,t-1), \tag{33}$$

$$\tilde{B}^1_{slv}(n,t) = B^2_{slv}(n,t-1). \tag{34}$$

Final state probabilities.

$$B^0_{slv}(0,t) = \tilde{B}^0_{slv}(0,t)C^0_{slv}(n,t), \tag{35}$$

$$B^1_{slv}(n,t) = \tilde{B}^0_{slv}(0,t)C^1_{slv}(n,t) + \tilde{B}^1_{slv}(n,t)C^0_{slv}(*,t), \tag{36}$$

$$B^2_{slv}(n,t) = \tilde{B}^0_{slv}(0,t)C^2_{slv}(n,t) + \tilde{B}^1_{slv}(n,t)(C^1_{slv}(*,t) + C^2_{slv}(*,t)), \tag{37}$$

$$B^0_{slv}(0,t) + B^1_{slv}(n,t) + B^2_{slv}(n,t) = 1. \tag{38}$$

3.3. IBC Buffers

The state transition probabilities for phase 1 of an IBC buffer are the same as those of the internal stage buffers. The state probabilities at the end of phase 2 of clock cycle t are described by

$$\boldsymbol{B}_{0lv}(n,t) = \tilde{\boldsymbol{B}}_{0lv}(n,t)\boldsymbol{C}''_{0lv}, \tag{39}$$

$$\boldsymbol{B}_{0lv}(b,t) = \tilde{\boldsymbol{B}}_{0lv}(b,t)\boldsymbol{C}'_{0lv}, \tag{40}$$

where $\boldsymbol{C}'_{0lv}$ and $\boldsymbol{C}''_{0lv}$ are the matrices of sizes $m \times m$ and $m \times (m+1)$, respectively;

$$\boldsymbol{C}''_{0lv} = \begin{bmatrix} 1-\lambda & \lambda & 0 & \dots & 0 \\ 0 & 1-\lambda & \lambda & \dots & 0 \\ 0 & 0 & 1-\lambda & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & \lambda \end{bmatrix} \quad \text{and} \quad \boldsymbol{C}'_{0lv} = \begin{bmatrix} 1-\lambda & \lambda & 0 & \dots & 0 \\ 0 & 1-\lambda & \lambda & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & \lambda \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}.$$

For example, the state probabilities of an IBC of size two are as follows.

Final state probabilities.

$$B^0_{0lv}(0,t) = \tilde{B}^0_{0lv}(0,t)(1-\lambda), \tag{41}$$

$$B^1_{0lv}(n,t) = \tilde{B}^0_{0lv}(0,t)\lambda + \tilde{B}^1_{0lv}(n,t)(1-\lambda), \tag{42}$$

$$B^2_{0lv}(n,t) = \tilde{B}^1_{0lv}(n,t)\lambda, \tag{43}$$

$$B^1_{0lv}(b,t) = \tilde{B}^1_{0lv}(b,t)(1-\lambda), \tag{44}$$

$$B^2_{0lv}(b,t) = \tilde{B}^1_{0lv}(b,t)\lambda + \tilde{B}^2_{0lv}(b,t), \tag{45}$$

$$B^0_{0lv}(0,t) + B^1_{0lv}(n,t) + B^2_{0lv}(n,t) + B^1_{0lv}(b,t) + B^2_{0lv}(b,t) = 1. \tag{46}$$

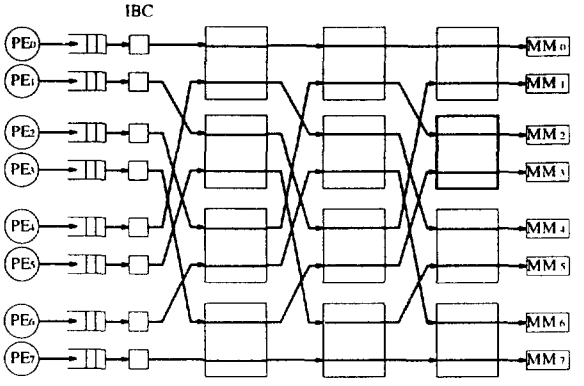


Fig. 3. An 8 × 8 Omega network with only SSSD traffic.

3.4. Throughput and delay

The throughput at the output slv is given by the number of packets leaving slv.

$$\mu_{slv} = 1 - \lim_{t\to\infty} B^0_{slv}(0,t). \tag{47}$$

For uniform traffic, $\mu_{slv} = \mu$ for all l and v. The packet loss probability for a uniform traffic is therefore given by

$$\eta = \frac{\lambda - \mu}{\lambda}. \tag{48}$$

Packet delay is defined to be the number of clock cycles a packet takes to reach the destination port from the source port. Let δ_k be the packet delay in the buffer of an SE in stage k.

$$\delta_k = \lim_{t\to\infty} \frac{\sum_{i=1}^{m} i\{B^i_{klv}(n,t) + B^i_{klv}(b,t)\}}{\sum_{i=1}^{m}\{B^i_{klv}(n,t)r_{klv}(n,t) + B^i_{klv}(b,t)r_{klv}(b,t)\}} \tag{49}$$

and the delay at IBC buffer is given by

$$\delta_{IBC} = \lim_{t\to\infty} \frac{\sum_{i=1}^{f} i\{B^i_{0lv}(n,t) + B^i_{0lv}(b,t)\}}{\sum_{i=1}^{f}\{B^i_{0lv}(n,t)r_{0lv}(n,t) + B^i_{0lv}(b,t)r_{0lv}(b,t)\}}. \tag{50}$$

Then, the total delay in the network is given by

$$\delta = \delta_{IBC} + \sum_{k=1}^{s} \delta_k. \tag{51}$$

Since the equations describing the dynamics of the network are described by recurrence relations, the solution is obtained by an iterative method [1].

4. Traffic patterns

To illustrate the accuracy of our proposed model, we have considered the following three types of traffic patterns in this study:

(1) Uniform, where the output destination of packets are uniformly distributed over all the outputs of the MIN.
(2) Single Source to Single Destination (SSSD) embedded in a uniform traffic pattern.
(3) An SSSD (ASSSD) path embedded in a uniform traffic pattern.

SSSD and ASSSD are nonuniform traffic patterns and are described below.

4.1. SSSD traffic pattern

The SSSD is a typical nonuniform traffic pattern and includes the uniform traffic pattern as its special case. It is obtained by adding SSSD traffic patterns on top of a uniform background traffic reference as shown in Fig. 3. It can reflect not only the favorite memory access application in a multiprocessor environment, but also some current and future ISDN applications in the communication networking area [30]. In a shared memory multiprocessor system, a processor is likely to address a particular memory module most of the time except when an interprocessor communication is needed. If processor PE_i communicates more often with memory module MM_i, MM_i will be called the favorite memory of PE_i. Assume that λ_{sssd} is the probability of the SSSD traffic from PE_i to MM_i. Therefore, the probability that PE_i requests MM_i is given by $\lambda_{sssd} + \lambda_U/N$, where λ_U is the rate of uniform traffic from PE_i to all the memory modules including MM_i. Then $\lambda = \lambda_{sssd} + \lambda_U$ is the offered traffic load at an input. The $N \times N$ load matrix for this traffic is given by

$$\begin{bmatrix} \lambda_{sssd} + \frac{\lambda_U}{N} & \frac{\lambda_U}{N} & \frac{\lambda_U}{N} & \cdots & \frac{\lambda_U}{N} \\ \frac{\lambda_U}{N} & \lambda_{sssd} + \frac{\lambda_U}{N} & \frac{\lambda_U}{N} & \cdots & \frac{\lambda_U}{N} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \frac{\lambda_U}{N} & \frac{\lambda_U}{N} & \frac{\lambda_U}{N} & \cdots & \lambda_{sssd} + \frac{\lambda_U}{N} \end{bmatrix}.$$

The static routing probabilities in the different stages are obtained as in [20] and are given below.

$$P_{kl\underline{uv}} = P_{kluv} = \frac{\lambda_{sssd} + 2^{(k-1)} \times 0.5\lambda_U}{2^{(k-1)}\lambda_U + \lambda_{sssd}}, \tag{52}$$

$$P_{kl\underline{uv}} = P_{kluv} = 1 - P_{kl\underline{uv}}. \tag{53}$$

4.2. An SSSD path embedded in a uniform traffic pattern

We investigate the influence of an SSSD traffic (ASSSD) path embedded in a uniform traffic pattern (Fig. 4). The traffic load from PE_0 to MM_0 is defined as λ_{asssd}. The Banyan network with input buffers has been studied under this traffic pattern in [16]. The $N \times N$ load matrix for Fig. 4 is given by

$$\begin{bmatrix} \lambda_{asssd} & 0 & 0 & \cdots & 0 \\ 0 & \frac{\lambda}{N-1} & \frac{\lambda}{N-1} & \cdots & \frac{\lambda}{N-1} \\ 0 & \frac{\lambda}{N-1} & \frac{\lambda}{N-1} & \cdots & \frac{\lambda}{N-1} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \frac{\lambda}{N-1} & \frac{\lambda}{N-1} & \cdots & \frac{\lambda}{N-1} \end{bmatrix}.$$

Following the approach of [20], the static routing probabilities corresponding to the load matrix given above can be shown to be:

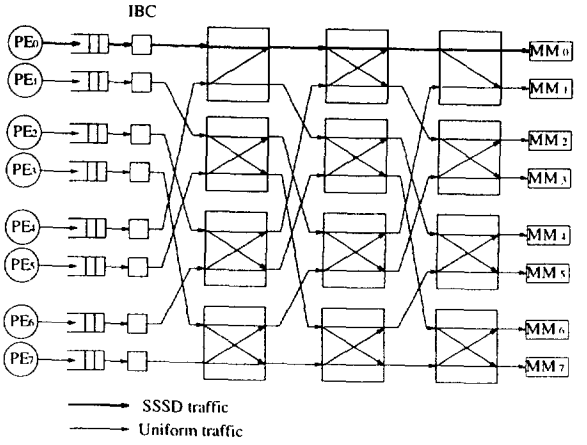


Fig. 4. An 8 × 8 Omega network with ASSSD traffic embedded in a uniform traffic pattern.

For $1 \le k \le n$, $l = 0$,

$$P_{kl\underline{x}y} = \frac{(2^{k-1}-1)(2^{s-k}-1)\frac{\lambda}{s-1} + \lambda_{\text{asssd}}}{(2^{k-1}-1)(2^{s-k}-1)\frac{\lambda}{s-1} + \lambda_{\text{asssd}} + (2^{k-1}-1)2^{s-k}\frac{\lambda}{s-1}}, \tag{54}$$

$$P_{kl\underline{x}y} = 1 - P_{kl\underline{x}y}, \tag{55}$$

$$P_{kl\underline{x}y} = \frac{2^{s-k}-1}{2^{s-k+1}-1}, \tag{56}$$

$$P_{kl\underline{x}y} = 1 - P_{kl\underline{x}y}. \tag{57}$$

For $1 \le k \le s-1$, $l = 2^{k-1} \times 1, 2^{k-1} \times 2, \ldots, 2^{k-1} \times \text{mod}((N/2-1)/2^{k-1})$,

$$P_{klu\underline{v}} = \frac{2^{s-k}-1}{2^{s-k+1}-1}, \tag{58}$$

$$P_{kluv} = 1 - P_{klu\underline{v}}. \tag{59}$$

The packets in the rest of the SEs are uniformly distributed.

5. Results

In this section, we investigate the accuracy of the model described in Section 3. The results from the basic model [21] (which is similar to the analysis in [2] when the network is operated under uniform traffic) and the proposed model are compared to those from simulations for various buffer sizes, network sizes and traffic conditions. The simulation methodology is described in [21,31,32]. In all the figures, except Fig. 16, the SE buffer size (m) and the IBC buffer size (f) are the same.

Fig. 5 shows the normalized throughput and delay of a two buffered Omega network of size 64 × 64 under uniform traffic. It is found that the accuracy of all the models are very good as long as the offered traffic load is small. However, the basic model is inaccurate under heavy offered traffic load because it does not take a rigorous account of blocked packets. The proposed model produces very accurate results because it accurately models the behavior of blocked packets. Fig. 6 shows the normalized throughput versus the SSSD and ASSSD traffic pattern. Since blocked packets are taken into account, the results from the proposed model are closer

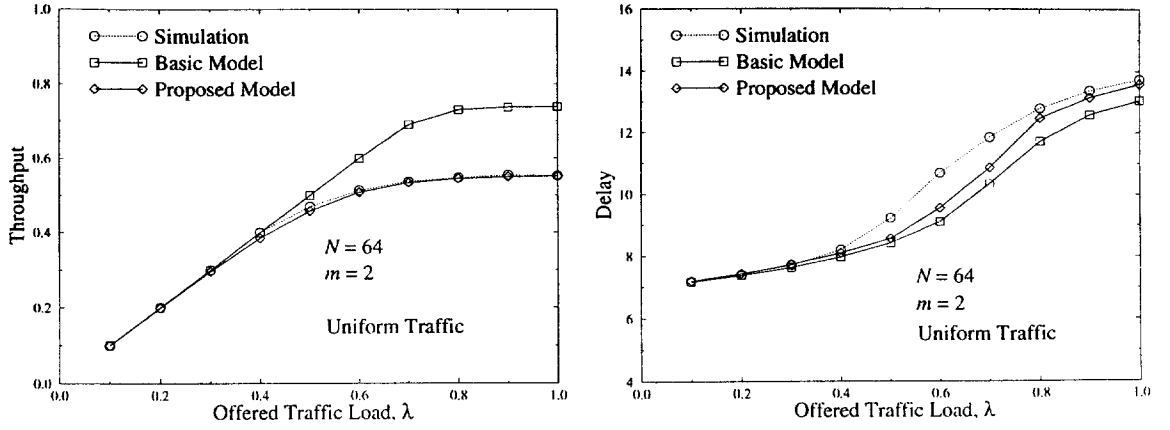


Fig. 5. Throughput and delay versus traffic load under uniform traffic for $N = 64$, $m = 2$.

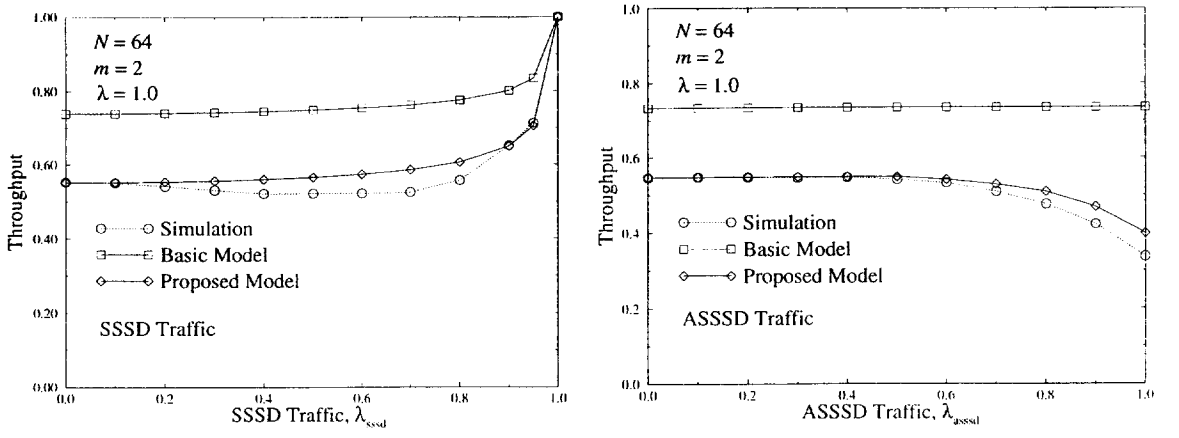


Fig. 6. Throughput versus SSSD and ASSSD traffic $N = 64$, $m = 2$.

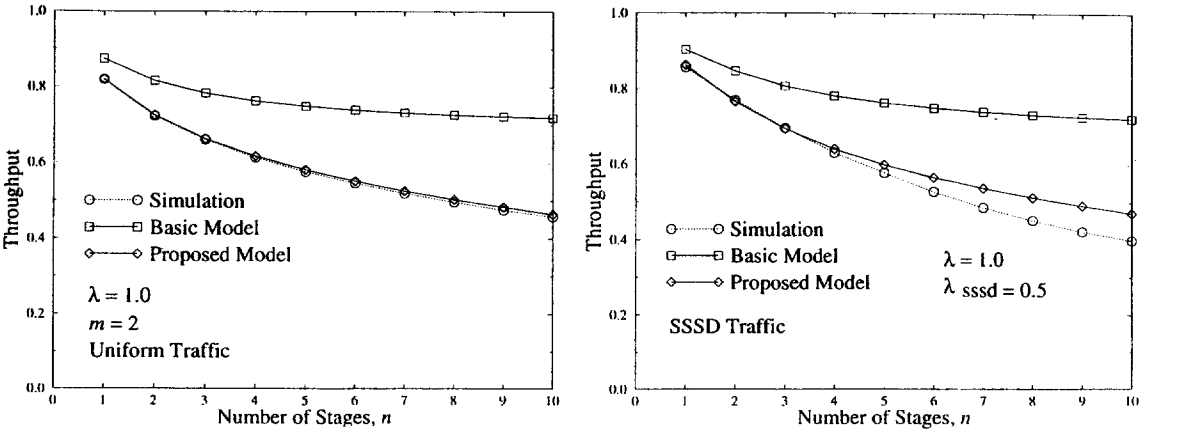


Fig. 7. Throughput versus network size for uniform and SSSD traffic.

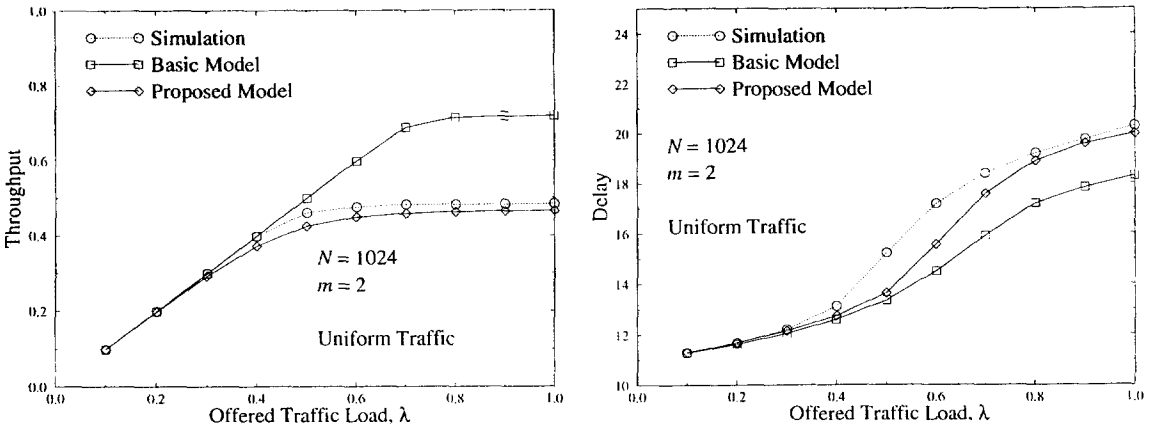


Fig. 8. Throughput and delay for uniform traffic in a large network.

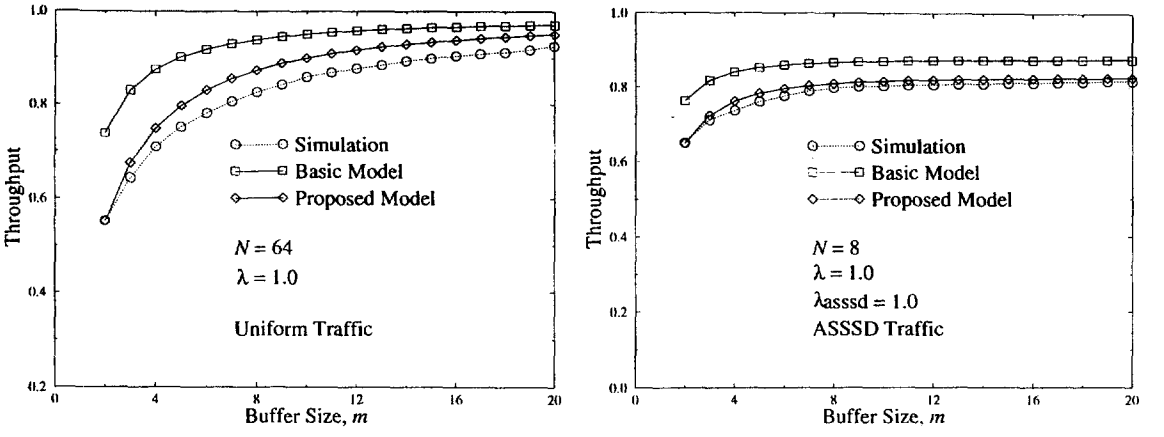


Fig. 9. Throughput versus buffer size for uniform and ASSSD traffic.

to simulation than those from the basic model. The influence of the size of the network on the performance is shown in Fig. 7. Since every additional stage of the network introduces further collisions, the normalized throughput decreases when the size of the network is increased. It is seen that the basic model is much less accurate than the proposed model. Fig. 8 shows the performance of the model when the network size is large ($N = 1024$) and operating under a uniform traffic. Note that the results from the proposed model are still close to those from simulation.

Fig. 9 shows the effect of buffer size on the throughput under uniform and ASSSD traffic patterns. Fig. 10 shows the effect of buffer size on the throughput and delay under the SSSD traffic. It is seen that the proposed model is much more accurate than the basic model, and buffer sizes greater than eight do not result in a significant increase in the throughput. Therefore, we will use buffers of size six in the rest of the figures.

Fig. 11 shows the normalized throughput and delay under a uniform traffic for $m = 6$. The proposed model shows better results than the basic model. Normalized throughput and delay of a 64×64 MIN under ASSSD traffic are shown in Fig. 12. Fig. 13 shows the effect of the buffer size on the throughput and delay for the ASSSD traffic. As expected, the proposed model shows much better results than basic model. Fig. 14 shows the normalized throughput and delay for a large network size ($N = 1024$) under SSSD traffic. Note that λ_{sssd}

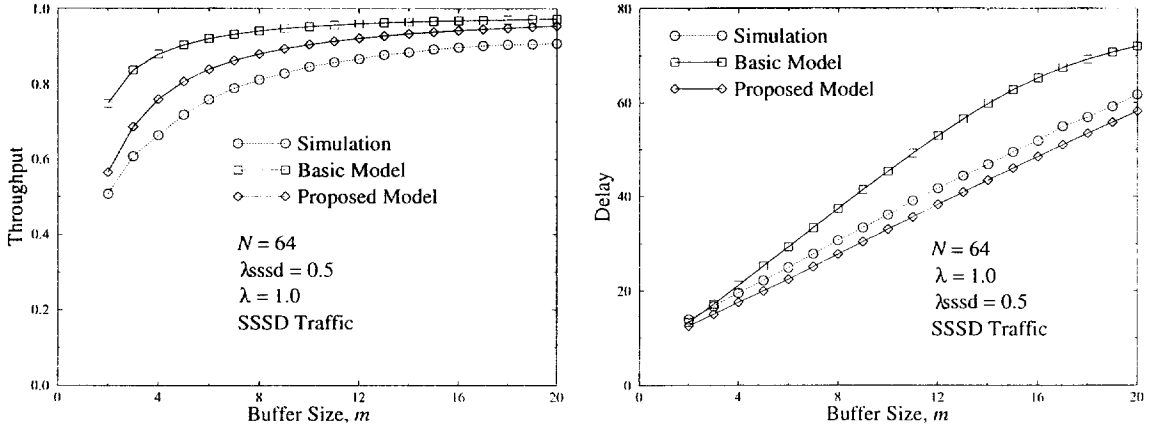


Fig. 10. Throughput and delay versus buffer size for SSSD traffic.

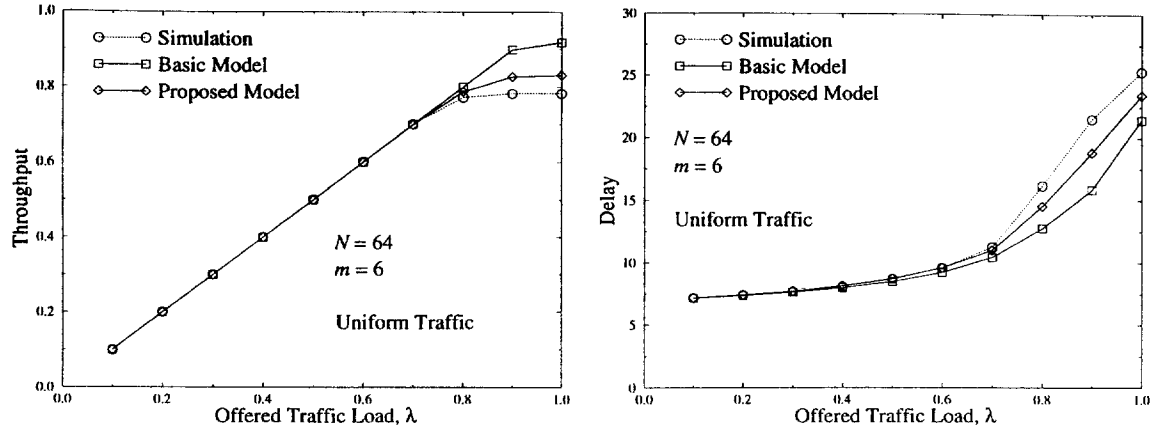


Fig. 11. Throughput and delay versus uniform traffic load for $N = 64$, $m = 6$.

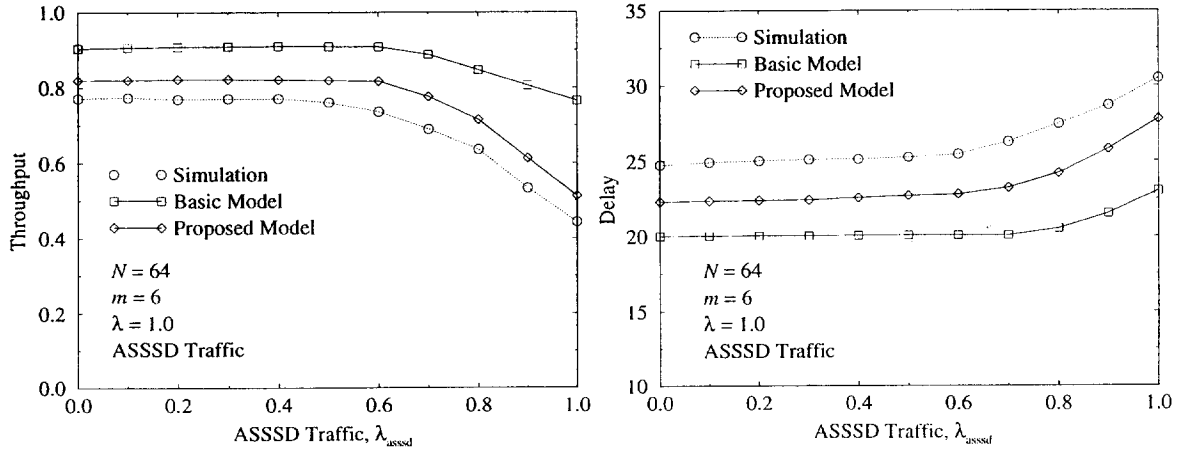


Fig. 12. Throughput and delay versus ASSSD traffic for $N = 64$, $m = 6$.

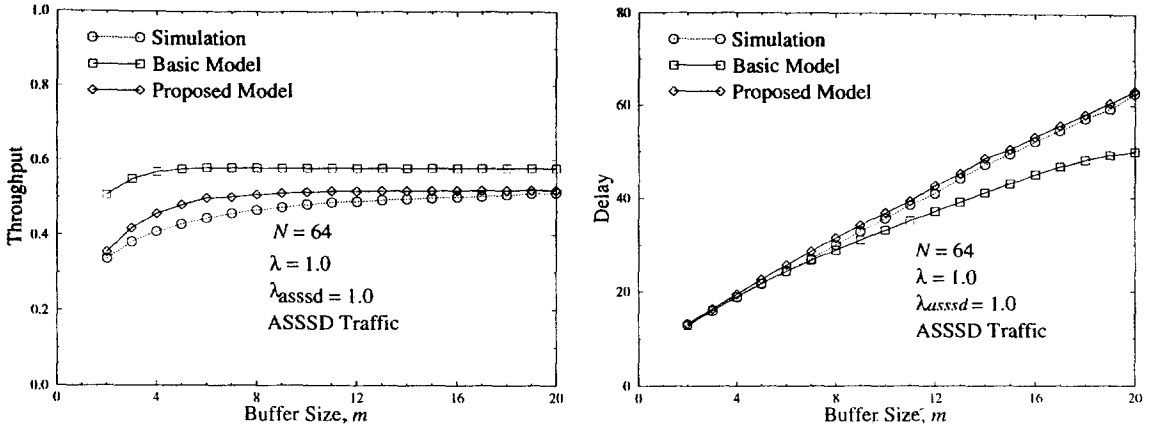


Fig. 13. Throughput and delay versus buffer size for ASSSD traffic for $N = 64$.

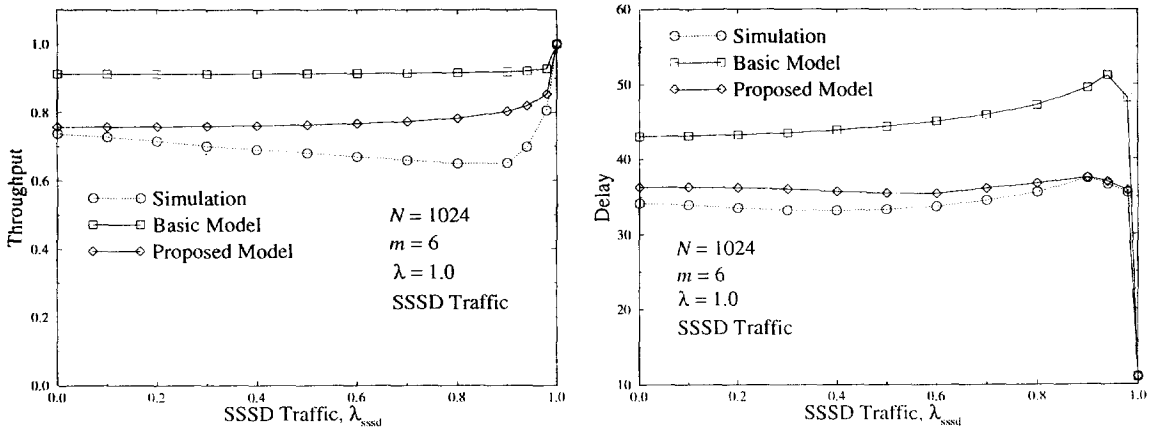


Fig. 14. Throughput and delay versus SSSD traffic for $N = 1024$, $m = 6$.

$= 0$, corresponds to a uniform traffic. Throughput and delay decrease with an increase in λ_{sssd} until $\lambda_{sssd} = 0.9$ when the throughput increases, and $\lambda_{sssd} = 0.6$ when the delay increases. When $\lambda_{sssd} = 1$, the uniform traffic becomes zero, the throughput is one and the delay drops to minimal. The effect can be explained by the fact that there is no conflict arising for SSSD traffic. Increased offered λ_{sssd} traffic causes more packet blocking in the SEs under lower λ_{sssd} and high uniform background load. But when λ_{sssd} load increases, the conflict between different inputs in an SE is reduced, and no conflict arises at $\lambda_{sssd} = 1$. i.e., there are two factors which are influencing the performance of the switch. First, the packet blocking due to the uniform traffic, which has a negative effect on the throughput. The second factor is that there is no internal routing conflict inside the switch in the case of purely SSSD traffic, which has a positive effect on the throughput. The net result of the above two factors determines the normalized throughput and mean delay.

Fig. 15 plots the effect of buffer size on the packet loss probability in 64×64 and 1024×1024 networks, obtained from the model, for several values of uniform traffic load. When the offered load is 0.4 for a 64×64 network, packet loss can be kept low (below 10^{-7}) if the buffer size is 7 or more. When the offered load is 0.8, the buffer size has to be increased to 20 to keep the packet loss at 10^{-7}. If the packet loss has to be kept low, the buffer size has to be increased to 20 when the offered load is 0.8. However, if the input load is greater

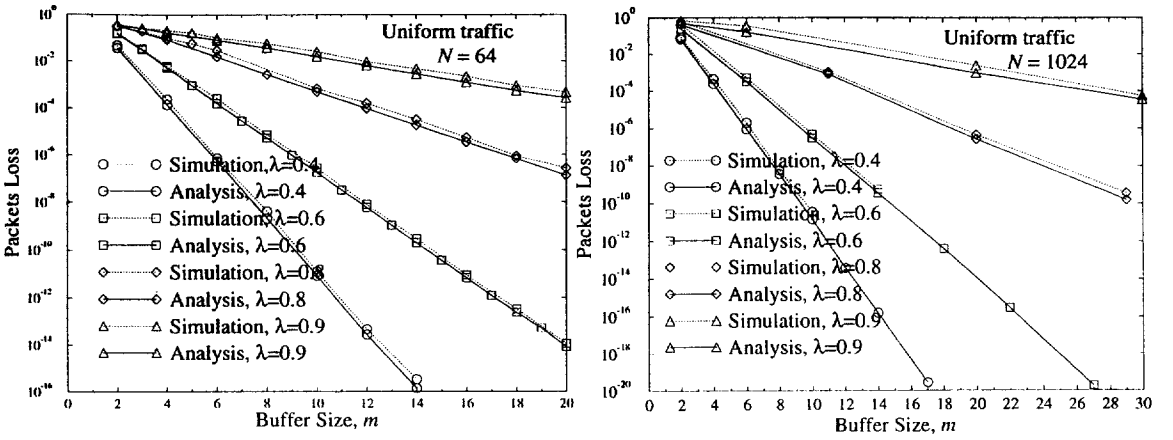


Fig. 15. Packets loss versus buffer size at different traffic loads for $N = 64$ and 1024. (Modeling and simulation results are indicated by solid and dashed lines, respectively.)

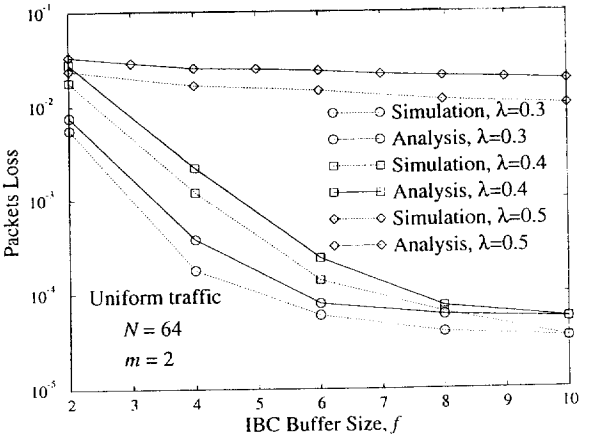


Fig. 16. Packets loss versus IBC buffer size at different traffic loads.

than 0.8 and the buffer size is smaller than 20, such a low packet loss can not be obtained. If the network size is increased to 1024 × 1024, the buffer size almost remains the same for a packet loss of 10^{-7} when the input load is 0.8. To maintain a packet loss of 10^{-4} with an input load of 0.9, the buffer sizes should be 20 and 30 for 64 × 64 and 1024 × 1024 networks respectively. Fig. 16 plots the packet loss probability as a function of the IBC buffer size f, for several values of offered load λ. The SEs are assumed to have two buffers. It shows that when the offered traffic load $\lambda = 0.3$ and 0.4, the packet loss asymptotically decreases with an increase in the IBC buffer size. However, increase in the IBC buffer size has no effect when the offered traffic load is higher than 0.5 because of the internal buffers become saturation.

6. Conclusions

A new analytic model for performance evaluation of MINs using finite output-buffered SEs has been introduced in this paper. Previous models either assumed uniform traffic or are inaccurate when the input traffic load is high. Analytical results obtained from the proposed model have been compared with those from an

existing model and simulation. The proposed model produces better results for both uniform and nonuniform traffic patterns. The novelty of the model lies in its ability to account for the blocking behavior of packets, the correlation of packet movement between consecutive network cycles. It has been shown that the states of buffers in two adjacent stages need to be carefully taken into consideration in order to obtain accurate results from Markov chain-based analytical models for MINs. The proposed model is general enough to analyze MINs with arbitrary buffer sizes and non-uniform traffic patterns. The basic idea contributing to the accuracy of the model may be used by other researchers in multistage interconnection networks.

The proposed model is solved iteratively. The model usually converges within 100 iterations when the error tolerance between successive iterations is set to 10^{-7}. For a 1024×1024 MIN, a SUN Sparcstation requires about 140 hours of simulation time as compared to about 2 hours for the proposed model. The proposed model is therefore much more computationally efficient than simulation.

The proposed model assumes a Bernoulli arrival processes for incoming packets. Real traffic in ATM environment is likely to be bursty in nature. However, modeling the performance of multistage switches under bursty traffic appears to be non-trivial. A challenging extension of this work would be to develop a model for a bursty traffic pattern. The proposed model could also be extended to allow modeling of MINs using $a \times a$, $a \geq 3$, SEs. Since we trace the possible states of the buffers in each SE, the number of states in the Markov chain grows exponentially with the number of input lines of an SE. Thus the generalization of the model to the case of $a \times a$ is difficult, especially under a bursty traffic pattern.

Acknowledgment

The authors would like to thank the anonymous reviewers for their many useful comments and suggestions. The reviewers' thorough and insightful comments greatly improved the quality of the paper.

References

[1] Y.C. Jenq, Performance analysis of a packet switch based on single-buffered banyan network, *IEEE J. Select. Areas Commun.* **1** (1983) 1014.

[2] H.S. Kim, I. Widjaja and A. Leon-Garcia, Performance of output-buffered banyan networks with arbitrary buffer sizes, in: *IEEE INFOCOM '91: Conf. on Computer Communications*, Bal Harbour, FL (1991) 701.

[3] B. Zhou, K.E. Forward and G.J. Armitage, Simulation study of the interaction between a multi-media terminal and the ATM network, *J. Electrical and Electronics Engineering* **13** (1) (1993) 41.

[4] G. Bianchi and J.S. Turner, Improved queueing analysis of shared buffer switching networks, *IEEE/ACM Trans. Networking* **1** (4) (1993) 482.

[5] S. Gianatti and A. Pattavina, Performance analysis of ATM banyan networks with shared queueing – Part I: Random offered traffic, *IEEE/ACM Trans. Networking* **2** (1994) 398.

[6] A. Pattavina and S. Gianatti, Performance analysis of ATM banyan networks with shared queueing – Part II: Correlated/unbalanced offered traffic, *IEEE/ACM Trans. Networking* **2** (1994) 411.

[7] P. Goli and V. Kumar, Performance of a crosspoint buffered ATM switch fabric, in: *IEEE INFOCOM'92* (1992) 426.

[8] P-Y. Chen, D.H. Lawrie, P-C. Yew and D.A. Padua, Interconnection networks using shuffles, *IEEE Comput.* **14** (12) (1981) 55.

[9] H. Ahmadi and W.E. Denzel, A survey of modern high-performance switching techniques, *IEEE J. Selected Areas Commun.* **7** (7) (1989) 1091.

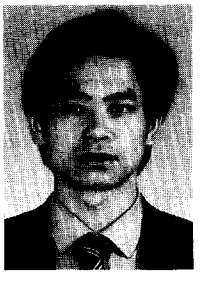
[10] B. Zhou and M. Atiquzzaman, A performance comparison of buffering schemes for multistage switches, in: *ICA3PP'95, 1st Internat. Conf. on Algorithms And Architectures for Parallel Processing*, Brisbane, Australia (1995) 809.

[11] I. Widjaja and A. Leon-Garcia, The StarBurst switch with dedicated output buffering and shared input buffering, in: *IEEE Internat. Conf. on Communications*, Geneva (1993) 691.

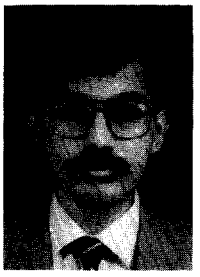
[12] H. Yoon, K.Y. Lee and M.T. Liu, Performance analysis of multibuffered packet-switching networks in multiprocessor systems, *IEEE Trans. Commun.* **39** (3) (1990) 319.

[13] T.H. Theimer, E.P. Rathgeb and M.N. Huber, Performance analysis of buffered banyan networks, *IEEE Trans. Commun.* **39** (2) (1991) 269.

[14] S.H. Hsiao and C.Y.R. Chen, Performance analysis of single-buffered multistage interconnection networks, *IEEE Trans. Commun.* **42** (9) (1994) 2722.
[15] Y. Mun and H.Y. Youn, Performance analysis of finite buffered multistage interconnection networks, *IEEE Trans. Comput.* **43** (2) (1994) 153.
[16] H.S. Kim and A.L. Garcia, Performance of buffered banyan networks under nonuniform traffic patterns, *IEEE Trans. Commun.* **38** (5) (1990) 648.
[17] M. Atiquzzaman and M.S. Akhtar, Effect of nonuniform traffic on the performance of multistage interconnection networks, in: *IEE Proc. Comput. Digit. Tech.* **141** (3) (1994) 169.
[18] S.C. Liew, Performance of input-buffered and output-buffered ATM switches under bursty traffic: Simulation study, in: *Globecom 90: IEEE Global Telecommunications Conf.* (1990) 1919.
[19] V.L. James and C.Y.R. Chen, An analytical model for partially blocking finite-buffered switching networks, *IEEE/ACM Trans. Networking* **2** (5) (1994) 533.
[20] T. Lin and L. Kleinrock, Performance analysis of finite-buffered multistage interconnection networks with a general traffic pattern, in: *ACM SIGMETRICS Conf. on Measurement and Modeling of Computer Systems* (1991) 68.
[21] B. Zhou and M. Atiquzzaman, Performance of output-multibuffered multistage interconnection networks under general traffic patterns, in: *IEEE INFOCOM '94: Conf. on Computer Communications*, Toronto, Canada (1994) 1448.
[22] B. Zhou and M. Atiquzzaman, Performance analysis of finite output-buffered multistage ATM switching fabrics, in: *IEEE INFOCOM'95* (1995) 16.
[23] B. Zhou and M. Atiquzzaman, Accurate analysis of multistage interconnection networks using finite output-buffered switching elements, in: *ICA3PP'95, 1st Internat. Conf. on Algorithms And Architectures for Parallel Processing*, Brisbane, Australia (1995) 838.
[24] O.E. Percus and S.R. Dickey, Performance analysis of clock-regulated queues with output multiplexing in three different 2×2 crossbar switch architectures, *J. Parallel Distributed Comput.* **16** (1) (1992) 27.
[25] B. Zhou and M. Atiquzzaman, Performance ATM switch fabric using cross-point buffers, in: *Asia-Pacific Conf. on Communications*, Osaka, Japan (1995).
[26] M. Saleh and M. Atiquzzaman, Queueing analysis of shared buffer switches for ATM networks, in: *GLOBECOM 94: IEEE Global Telecommunications Conf.*, San Francisco, CA (1994) 1070.
[27] J.H. Patel, Performance of processor-memory interconnection for multiprocessors, *IEEE Trans. Comput.* **30** (10) (1981) 771.
[28] C. Kruskal and M. Snir, The performance of multistage interconnection networks for multiprocessors, *IEEE Trans. Comput.* **32** (1983) 1091.
[29] G.F. Pfister and V.A. Norton, Hot spot contention and combining in multistage interconnection networks, *IEEE Trans. Comput.* **34** (10) (1985) 943.
[30] J. Ding, Nonuniform traffic analysis of multistage interconnection networks with split buffers, in: *IEEE ICC'93*, Geneva, Switzerland (1993) 58.
[31] B. Zhou and M. Atiquzzaman, Performance of output-multibuffered multistage interconnection networks under nonuniform traffic patterns, in: *Internat. Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'94)* (1994) 405.
[32] B. Zhou and M. Atiquzzaman, Impact of switch architectures on the performance of multistage interconnection networks, in: *IEEE TENCON: Region 10's 9th Ann. Internat. Conf.*, Singapore (1994) 365.



Bin Zhou received the B.E. from the Harbin Institute of Technology, P.R. China in 1982 and the M.Sc.E. from the University of Melbourne, Australia in 1992, both in Electrical and Electronic Engineering. From 1982 to 1989, he was an engineer in the Communication Technology Research Institute, P.R. China. He has completed his Ph.D. program at Monash University. He has been with the Defence Communications Research Centre, the Institute for Telecommunications Research, University of South Australia. Currently he is with Motorola Australia Software Centre. His research interests include B-ISDN, ATM networks, and adaptive radio networks.



M. Atiquzzaman received the M.Sc. and Ph.D. degrees in elctrical engineering and electronics from the University of Manchester Institute of Science and Technology, England in 1984 and 1987, respectively. He has been with the Department of Electrical Engineering at King Fahd University of Petroleum & Minerals, Saudi Arabia end the Department of Computer Science and Computer Engineering at La Trobe University, Melbourne, Australia. Currently he is an academic staff member in the Department of Electrical & Computer Systems Engineering at Monash University, Clayton, Australia. His current research interests are in B-ISDN and ATM networks, multiprocessor systems, interconnection networks, parallel processing, image processing.