Performance Analysis of General Cut-Through Switching on Buffered MIN Switches*

CHAN L. LIAO AND WOEI LIN
Institute of Computer Science
National Chung Hsing University
Taichung, 402 Taiwan
E-mail: wlin@cs.nchu.edu.tw

This paper presents a general analytical model for studying the effect of cut-through switching on buffered MIN (multistage interconnection network) switches. We consider two types of MIN switches: input-buffered MIN and output-buffered MIN. Previous studies of cut-through switching assumed the use of 2×2 switching elements as a building block for the MIN; and they are not applicable to MINs using switches of other sizes than 2×2. With the general model, we find that switch size actually plays a key role in determining the performance of cut-through switching on the buffered MIN switch. Our proposed model and analysis successfully exhibit these performance characteristics, many of which were unknown previously, in terms of normalized throughput and delay under various operating conditions. The proposed model allows us to quantitatively analyze the joint effect of major system parameters, switch size and others, on the performance of cut-through switching for the two types of MINs. Our analysis essentially shows that cut-through switching has different effects on the performance of the input-buffered MIN switch and the output-buffered MIN switch. The differences are indicated and elaborated. In addition, through simulation we study the performance of cut-through switching on the buffered MIN switch under bursty traffic and hot-spot traffic.

Keywords: high-speed switching, input/output buffers, cut-through switching, performance analysis, interconnection networks

1. INTRODUCTION

Cut-through is an effective switching technique combining circuit switching and packet switching for fulfilling performance requirements of QoS supports. It was first presented by Kermani and Kleinrock in the context of computer networks [1]. In this paper we consider using cut-through switching in buffered multistage interconnection networks (MINs). In the buffered MIN, when a packet arrives at a switching element, and if the outgoing link is free, it proceeds immediately to the next stage. It allows packets to skip unnecessary buffering at the intermediate stages, moving packets rapidly to the destinations. This makes the buffered MIN appealing as a high-speed switching fabric.

Many analyses have been done for characterizing the performance of buffered MINs. However, they often assume that the MIN operates in a store-and-forward mode.

Received September 20, 2001; accepted April 15, 2002.
Communicated by Jang-Ping Sheu, Makoto Takizawa and Myongsoon Park.
* This research is supported in part by National Science Council, Taiwan, R.O.C., NSC-88-2213-E-020.

Kruskal and Snir provided a performance analysis for MINs with infinite buffer capacity [2]. Dias and Jump used the timed Petri-net model to analyze the performance of MINs with single buffers. Jeng presented the first Markov-chain performance model for single-buffered MINs [3]. Szymanski and Shaikh analyzed the performance of banyan networks with arbitrary switches and queue sizes [4]. Yoon et al. presented an analytical model for the performance of multi-buffered packet-switched MINs [5].

Relatively little research has been devoted to the performance study of cut-through switching on buffered MINs. Bubenik and Turner pioneered a simulation study on synchronous MIN switches operating under virtual cut-through [6]. Widjaja et al. presented an elegant analytical method for performance analysis of cut-through MIN switches [7]. However, their method is restricted to MINs with 2×2 switching elements. There is difficulty in applying it to MINs with switching elements other than 2×2. Other relevant literature on cut-through switching can be found in [8-10, 14-16].

The objective of our research is to present a general analytical method better understand the performance of buffered MINs using cut-through switching techniques. The proposed model is different the previous ones in two respects. First, the proposed model allows us to analyze MINs constructed from switching elements of arbitrary size, with buffers of arbitrary length, and with input buffers or output buffers. Second, this model assumes that cut-through can take place if the desired outgoing link is free. This gives the input-buffered MIN more flexibility in using cut-through to transfer packets. With this model, we gain insight to many performance characteristics of the cut-through buffered MIN, which were previously impossible to observe. The remaining discussion is organized as follows. In section 2, we introduce the basic assumptions and parameters required for modeling the buffered MIN using cut-through switching. Section 3 presents the analytical model for cut-through switching on the buffered MIN. Closed forms of a variety of packet flows are derived. In section 4, we present the analytical results, and compare them with simulation results. Performance characteristics due to cut-through switching are examined. We also present some simulation results for bursty and hot-spot traffic. Finally, conclusions are drawn in section 5.

2. PRELIMINARIES

2.1 Assumptions

We analyze the buffered MIN using cut-through switching under uniform traffic with the following assumptions.

(1) At each source node packets are generated by a Poisson process with the same rate.
(2) Packet destinations are uniformly distributed.
(3) All packets are of the same length. The transmission time of the MIN is divided into equal-length time slots, called network cycles. The duration of a network cycle is long enough for determining routing path and transmitting a packet across stages using cut-through.
(4) Operation of the MIN network is synchronous. At the beginning of a network

cycle, availability of outgoing links is examined. Then a packet is moved as far as it can be via cut-through, and it is stored in the buffer at the end of the cycle.

(5) Packets in buffers are delivered on a first-come-first-serve basis. On every network cycle, only one packet is allowed to move across an output link of a switching element.

2.2 Network Structure and Operations

An $N \times N$ MIN considered here consists of n stages of $a \times a$ switching elements, where $n = log_aN$. Each stage comprises N/a switching elements. Packets are moved across stages by a well-known self-routing technique [11], which specifies a unique path from a network input to a network output in a distributed manner. Each switching element contains a set of buffers for storing packets that cannot be forwarded to the subsequent stage immediately. We consider two classes of buffered MINs: input-buffered MIN and output-buffered MIN. In the case of input-buffered MINs, one packet is chosen randomly for forwarding when two packets or more in a switching element compete for the same output link. The remaining packets stay in their buffers and compete again in the next network cycle. In the case of output-buffered MINs, packets from different inputs are stored in an output buffer when they arrive at the buffer at the same time. However, if the buffer does not have sufficient space for all the arriving packets, only some of them are accepted, up to the available space, and the remaining packets have to stay one stage back, waiting for the next network cycle.

Next, we intend to establish the condition for using cut-through switching to bypass unnecessary buffering in MINs. Let us consider two extreme cases. First, a packet always comes across a nonempty buffer or fails to gain a free link at every stage along the designated path. As a result, it has to make one stop at every stage. In this case, the packet is delivered exactly in the manner of packet switching. In the second case, a packet encounters an empty buffer or successfully gains a free link at every stage. It takes only one network cycle to forward the packet to its final destination in a way of circuit switching. Combining these two cases, we come up with a new routing requirement for carrying out cut-through switching in buffered MINs: when arriving at some stage, a packet can immediately move to the next stage without being buffered if the desired output port is available. This makes cut-through switching more effective for the buffered MIN. We assume that a backpressure mechanism exists between adjacent stages in order to signal potential cut-through. At the beginning of each network cycle, switching elements send backpressure signals to their respective upstream switching elements, reporting the buffer status. If the routing requirement is met, a cut through the buffer takes place and the packet immediately advances to the next stage without being buffered.

2.3 Notation

For convenience, we provide a summary of notation to be used in the development of the performance model and analysis.

a: Size of switching element
n: Number of stages in the network

N: Size of MIN

m: Size of buffer

$p_0(k, t)$: Probability that a buffer of a switching element in stage k is empty at network cycle t

$p_j(k, t)$: Probability that j packets are stored in a buffer of a switching element in stage k at network cycle t

$p_m(k, t)$: Probability that a buffer of a switching element in stage k becomes full at network cycle t

$r_b(k, t)$: Probability that a packet in a buffer of a switching element in stage k successfully moves to stage $k + 1$ at network cycle t

$r_c(k, t)$: Probability that a packet coming to stage k becomes a cut-through packet and successfully moves to stage $k + 1$ at network cycle t

$q_b(k, t)$: Probability that at least one packet from buffers of a switching element in stage $k - 1$ is destined to a designated input port of a switching element in stage k at network cycle t

$q_c(k, t)$: Probability that at least one cut-through packet in stage $k - 1$ is destined a designated input port of a switching element in stage k at network cycle t

$q(k, t)$: Probability that at least packet, a buffered packet or a cut-though packet, in stage $k - 1$ is destined a designated input port of a switching element in stage k at network cycle t

3. MODEL AND ANALYSIS

This section presents an analytical performance model for MINs using cut-through switching under uniform traffic. We base our performance model on a well-known research work for modeling the store-and-forward MIN [3, 5]. The work, in fact, is a first-moment approximation for the performance analysis of the store-and-forward MINs, yet with significant reduction in modeling complexity. Our approach to modeling the cut-through MIN is briefly described as follows. Taking advantage of the structural symmetry of the MIN, we decompose the network into individual stages and reduce each stage into a single switching element. The switching element is further simplified to an input or output port with a buffer represented by a Markov chain. We derive closed-form equations for the probabilities corresponding to various packet flows through the buffer. Then, flow rates between adjacent stages are matched. Finally, we obtain two equations for performance measures.

3.1 Cut-through With Input Buffers

The key to modeling the input-buffered MIN using cut-through switching lies in relating the probabilities of packet flows listed in the previous section. For clarity, Fig. 1 is given to illustrates the relations among these probabilities. Note that in a switching element with input buffers, an incoming packet to an input port can make a cut through the associated input buffer as long as the desired output port is available. On the other hand, this requires that all the head-of-line packets in the buffers must head for different output ports from the one requested by the incoming packet.

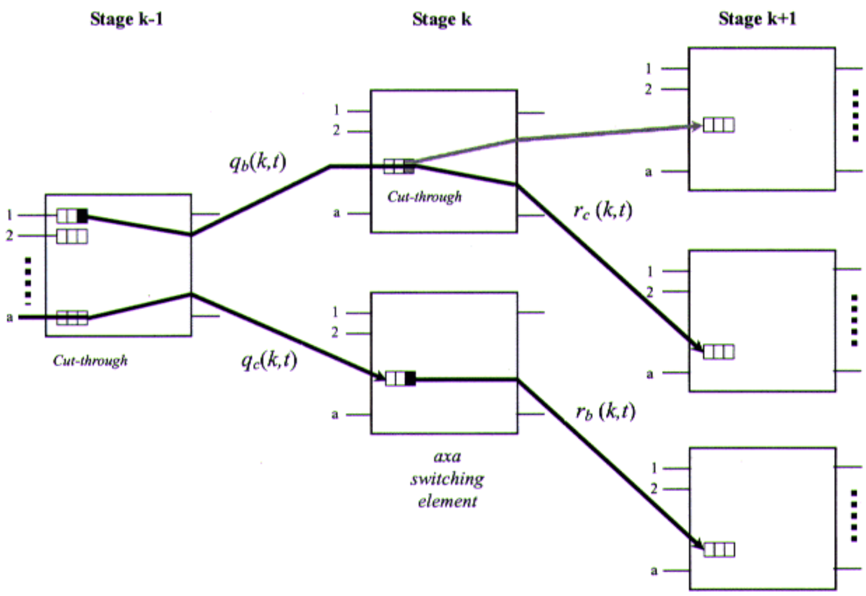

Fig. 1. Relation ships among probabilities for cut-through switching in input-buffered MINs.

We start with $q_b(k, t)$. It is the probability that at least one packet from the buffers of a switching element in stage $k - 1$ is destined a designated input port of a switching element in stage k at network cycle t, and is given by [12]

$$q_b(k, t) = 1 - [1 - \overline{p_0}\,(k - 1, t)/a]^a \quad 2 \le k \le n \tag{1}$$

Next, we would like to figure out $q_c(k, t)$. It is the probability that at least one cut-through packet from stage $k - 1$ is destined a designated input port of a switching element in stage k at network cycle t. It is expressed as

$$\begin{aligned} q_c(k,t) &= \sum_{j=0}^{a} \binom{a}{j} q(k-1,t)^j q_0(k-1,t)^{a-j} [1-(1-1/a)^j] \\ &= 1 - [1 - q(k - 1, t)/a]^a \quad 2 \le k \le n. \end{aligned} \tag{2}$$

In the equation above, $\binom{a}{j} q(k-1,t)^j q_0(k-1,t)^{a-j}$ is the probability that at the beginning of network cycle t a total of j packets arrive at a switching element at stage $k - 1$. $[1 - (1 - 1/a)^j]$ is the probability that, among the j packets, there exists at least one packet which heads for the designated input port of the succeeding switching element at stage k. Therefore, we have

$$\begin{aligned} q(k, t) &= q_b(k, t) + q_c(k, t) - q_b(k, t)q_c(k, t) \\ &= [1 - (1 - \overline{p_0}\,(k - 1, t)/a)^a(1 - q(k - 1, t)/a)^a] \quad 2 \le k \le n. \end{aligned} \tag{3}$$

We now show how $r_b(k,t)$ and $r_c(k,t)$are determined. $r_b(k,t)$ can be written as

$$r_b(k, t) = [q_b(k + 1, t)/ \overline{p_0}\,(k, t)][r_c(k + 1, t) + \overline{r_c}\,(k+1,t)\,\overline{p_m}\,(k+1,t) + \overline{r_c}\,(k + 1, t)\,p_m(k +1, t)r_b(k+1, t))] \quad 1\le k \le n - 1 \tag{4}$$

In the equation above, within the first pair of brackets is the probability that a packet is chosen among all the competing packets for passing a desired output port. Within the second pair of brackets, the first term accounts for the probability that a packet successfully moves out of the desired output port at stage k and becomes a cut-through packet at stage $k + 1$. The second term is the probability that a packet moves into a nonempty buffer at stage $k + 1$. The third term is the probability that a packet comes across a full buffer at stage $k + 1$ that happens to have a packet leaving immediately. Therefore, in this case the packet can still move into the buffer at stage $k + 1$. $r_c(k,t)$ can be expressed as

$$r_c(k, t) = \{q_c(k + 1, t)[1 - q_b(k + 1, t)]/q(k, t)\}[r_c(k + 1, t) + \overline{r_c}\,(k + 1, t)\,\overline{p_m}\,(k + 1, t) + \overline{r_c}\,(k + 1, t)\,p_m(k + 1, t)r_b(k + 1, t))] \quad 1 \le k \le n - 1 \tag{5}$$

In the equation above, the term enclosed by the curly braces is the probability that a packet arriving at stage k successfully makes its way toward the desired output port. This requires that (1) the packet be chosen out of the arriving packets that attempt to depart from this output port, and (2) none of the head-of-line packets in the buffers is heading for this output port. With these probabilities in mind, now we develop a state transition diagram of the buffer at stage k, as illustrated in Fig. 2.

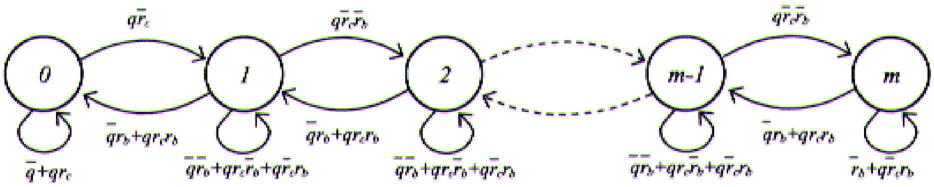

Fig. 2. State transition diagram for input buffer of depth m and with cut-through.

In the diagram, the buffer has $m + 1$ states, each corresponding to a specific number of packets in the buffer. The rates of packet flows are used in deriving the following equilibrium probabilities p_i of finding i, $0 \le i \le m$, packets in the buffer.

$$p_0(k, t + 1) = p_0(k, t)[\,\overline{q}\,(k, t) + q(k, t)r_c(k, t)] + p_1(k, t)[\,\overline{q}\,(k, t)r_b(k, t) + q(k, t)r_c(k, t)r_b(k, t)] \quad 1 \le k \le n \tag{6}$$

$$p_1(k, t + 1) = p_0(k, t)[q(k, t)\,\overline{r}_c\,(k, t)] + p_1(k, t)[\,\overline{q}\,(k, t)\,\overline{r}_b\,(k, t) + q(k, t)r_c(k, t)\,\overline{r}_b\,(k, t) + q(k, t)\,\overline{r}_c\,(k, t)r_b(k, t)] + p_2(k, t)[\,\overline{q}\,(k, t)r_b(k, t) + q(k, t)r_c(k, t)r_b(k, t)] \quad 1 \le k \le n \tag{7}$$

$$p_j(k, t + 1) = p_{j-1}(k, t)[q(k, t)\,\overline{r}_c\,(k, t)\,\overline{r}_b\,(k, t)] + p_j(k, t)[\,\overline{q}\,(k, t)\,\overline{r}_b\,(k, t) + q(k, t)r_c(k, t)\,\overline{r}_b\,(k, t) + q(k, t)\,\overline{r}_c\,(k, t)r_b(k, t)] + p_{j+1}(k, t)[\,\overline{q}\,(k, t)r_b(k, t) + q(k, t)r_c(k, t)r_b(k, t)] \quad 1 < j < m \text{ and } 1 \le k \le n \tag{8}$$

$$p_m(k, t + 1) = p_{m-1}(k, t)[q(k, t)\,\overline{r}_c\,(k, t)\,\overline{r}_b\,(k, t)] + p_m(k, t)[\,\overline{r}_b\,(k, t) + q(k, t)\,\overline{r}_c\,(k, t)r_b(k, t)] \quad 1 \le k \le n \tag{9}$$

The two performance measures of interest are as follows. Note that the two measures are normalized with repect to network size. Throughput is defined to be the probability that a packet successfully reaches the designated network output. Namely, it is equal to the steady-state value of $q(n + 1, t)$ and it is denoted as $q(n + 1)$. Other steady-state values are denoted as $q_b(n + 1)$, $q_c(n + 1)$ and $p_i(k)$.

$$\begin{aligned} Throughput_{input\text{-}buffer} &= q(n + 1) \\ &= q_b(n + 1) + q_c(n + 1) - q_b(n + 1)q_c(n + 1) \end{aligned} \tag{10}$$

Based on Little's law, we can calculate the normalized delay as

$$Delay_{input\text{-}buffer} = \frac{1}{n}\frac{1}{q(n+1)}\sum_{k=1}^{n}\sum_{i=1}^{m} ip_i(k). \tag{11}$$

3.2 Cut-through With Output Buffer

Now we consider output-buffered MINs using cut-through switching. In an output-buffered MIN, each output port of a switching element is preceded with a buffer. Cut-through switching operates differently in output-buffered MINs. In the following, we point out some operational differences for output-buffered MINs. First, let us consider an output port with an empty buffer. Packets entering a switch may compete for an output port at the same time. In this case, one of them is chosen arbitrarily to pass through the output port. The remaining packets are moved into the output buffer. Provided there are more packets than the buffer capacity, they are randomly selected to be stored in the buffer; the others have to step one stage back. Next, we consider an output port with a nonempty buffer. Several packets may simultaneously come to an output port with some existing packets in the buffer. In this case, these incoming packets yield the output port; and the head-of-line packet in the buffer gains access to the output port. In other words, no cut-through is performed. Some of the incoming packets up to the available buffer space are appended to the queue if the output buffer; the remaining packets simply back off one stage.

We proceed to figure out probabilities concerning packet flows between adjacent stages. Fig. 3 shows the logical association of these probabilities with the output buffers and links.

Here we introduce a new probability $r_{dj}(k, t)$. It is the probability that j packets come to some output buffer of a switching element at stage k at the beginning of cycle network t. Note that the probability that a packet reachs a designated output port is $q(k, t)/a$. Therefore, $r_{dj}(k, t)$ is given by

$$r_{dj}(k, t) = \binom{a}{j}[q(k, t)/\mathrm{a}]^j[1 - q(k, t)/a]^{a-j} \quad 1 \le k \le n \text{ and } 0 \le j \le a \tag{12}$$

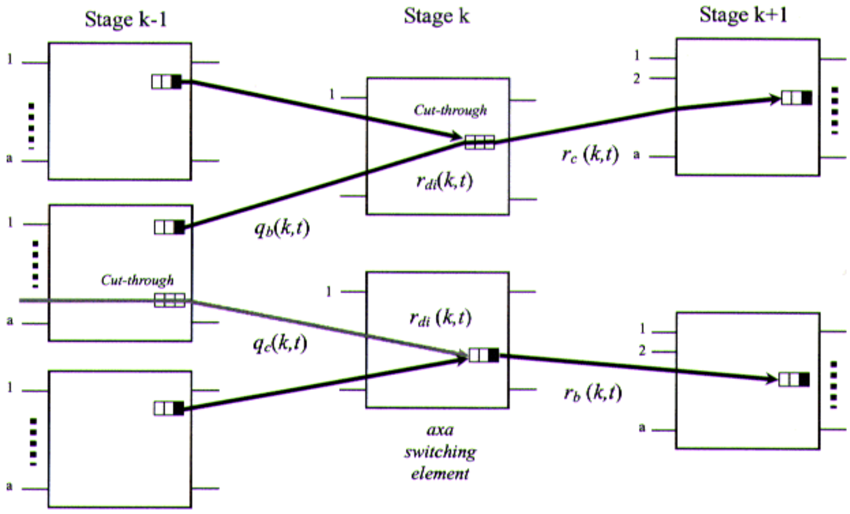

Fig. 3. Relation ship among probabilities for cut-through switching on output-buffered MINs.

Probabilities corresponding to various packet flows are expressed as follows.

$$q_b(k, t) = \overline{p_0}\ (k-1, t) = 1 - p_0(k-1, t) \quad 2 \le k \le n \tag{13}$$

$$q_c(k, t) = p_0(k-1, t)(1 - r_{d0}(k-1, t)) \quad 2 \le k \le n \tag{14}$$

$$\begin{aligned} q(k, t) &= q_b(k, t) + q_c(k, t) \\ &= 1 - p_0(k-1, t) + p_0(k-1, t)(1 - r_{d0}(k-1, t)) \\ &= 1 - p_0(k-1, t)r_{d0}(k-1, t) \quad 2 \le k \le n \end{aligned} \tag{15}$$

Now we examine $r_b(k, t)$, which is the probability that a buffered packet or a cut-through packet at stage k will advance successfully to stage $k + 1$. As mentioned earlier, there are two situations where a packet at stage k can advance: (1) it becomes a cut-through packet at the stage $k + 1$, or (2) it obtains a vacancy in the buffer at stage $k + 1$. Before calculating $r_b(k, t)$, we introduce T(k, t), the average number of packets a buffer at stage k can accept.

$$\begin{aligned} T(k, t) = {} & [p_0(k, t) + p_1(k, t)r_b(k, t)] \times [\sum_{i=1}^{m} ir_{di})(k,t) + \sum_{i=m+1}^{a}(mr_{di}(k,t)] + \\ & \sum_{j=1}^{m-1}[p_j(k,t)\bar{r}_b(k,t) + p_{j+1}(k,t)r_b(k,t)] \times [\sum_{i=1}^{m-j} ir_{di}(k,t) + \\ & \sum_{i=m-j+1}^{a}(m-j)r_{di}(k,t)] + p_0(k,t)[\sum_{i=m+1}^{a} r_{di}(k,t)] \quad 2 \le k \le n \end{aligned} \tag{16}$$

T(k, t) consists of three terms. The first term is the average number of packets a stage-k buffer can accept when it is empty or becomes empty due to an imminent packet departing at time t. The second term is the average number of packets a stage-k buffer can accept when it currently has j packets or $j + 1$ packets with a departing packet, $0 \le j \le m - 1$. The third term, $p_0(k,t)[\sum_{i=m+1}^{a} r_{di}(k,t)]$, accounts for an extra space left by a cut-through packet. It allows a buffer at stage k to accommodate an addition packet from stage $k - 1$. T(k, t), in fact, includes cut-through packets as a special case of buffered packets. Therefore, we consider $r_b(k, t)$ to be the gross rate of packet over an output link. Because the flow between stages k and $k + 1$ has to be conserved, the average number of packets leaving stage k is equal to the average number of packets accepted by stage $k + 1$; that is,

$$q(k, t)r_b(k, t) = \mathrm{T}(k + 1, t).$$

Hence, we have

$$r_b(k, t) = \mathrm{T}(k + 1, t)/q(k, t) \qquad (17)$$

Having all the packet-flow rates in place, we now present a state-transition diagram for the output-buffered MIN in Fig. 4. State transitions for output buffers become more complicated, because of the possibility of simultaneous packet arrivals. For simplicity, in the figure we show only transitions into a reference state j and a few relevant states. Let us calculate $p_j(k, t)$, the probability of finding j packets in a buffer of a switching element at stage k at the beginning of cycle $t + 1$. The transition probability from an interior state i to state j is $\overline{r_b}\, r_{d(j-i)} + r_b r_{d(j-i+1)}$, $1 \le j \le m - 1$ and $0 \le i \le j$. The transition probabilities from states 0 and $j + 1$ to state j are $r_b r_{d(j+1)}$ and $r_b r_{d0}$, respectively.

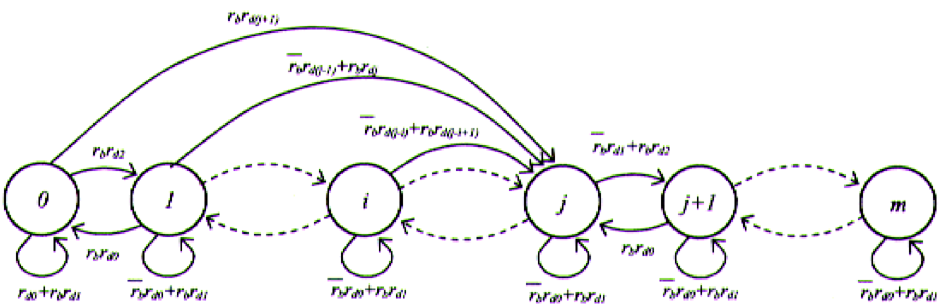

Fig. 4. State transition diagram for output buffer of depth m and with cut-through.

Therefore, $p_j(k, t + 1)$ is given as the sum of the three probabilities

$$p_j(k, t + 1) = p_0(k, t)r_b(k, t)r_{d(j+1)}(k, t) + \sum_{i=1}^{j}\{p_i(k,t)[\overline{r}_b(k,t)r_{d(j-i)}(k,t) + r_b(k, t)\, r_{d(j-i+1)}(k, t)]\} + p_{j+1}(k, t)r_b(k, t)r_{d0}(k, t) \quad 1 \le k \le n \text{ and } 1 \le j \le m-1 \qquad (18)$$

As to the probabilities of the two boundary states, we have

$$p_0(k, t + 1) = p_0(k, t)[r_{d0}(k, t) + r_b(k, t)r_{d1}(k, t)] + p_1(k, t)r_b(k, t)r_{d0}(k, t) \quad 1 \le k \le n \qquad (19)$$

$$p_m(k, t+1) = p_0(k, t)[\sum_{j=m+1}^{a} r_{dj}(k,t) + \sum_{i=1}^{m}\{p_i(k,t)[\bar{r}_b(k,t)\sum_{j=m-i}^{a} r_{dj}(k,t) + r_b(k,t)\sum_{j=m-i+1}^{a} r_{df}(k,t)]\} \quad 1 \le k \le n \quad (20)$$

Now that all the steady-state probabilities are available, the two performance measures – normalized throughput and delay, are

$$Throughput_{output\text{-}buffer} = 1 - p_0(n) + q(n)r_c(n) \quad (21)$$

$$Delay_{output\text{-}buffer} = \frac{1}{n}\frac{1}{1-p_0(n)+q(n)r_c(n)}\sum_{k=1}^{n}\sum_{i=1}^{m} ip_i(k) \quad (22)$$

4. COMPARISON AND SIMULATION

In this section, we quantitatively evaluate the performance of cut-through switching on input-buffered and output-buffered MINs, respectively. Of particular interest is to observe the performance improvement made by cut-through switching. We carry out the evaluation and present the results in terms of two performance measures, normalized throughput and delay, as a function of some system parameters under input buffering and output buffering conditions. Here we consider four major system parameters: switching element size (a), number of stages (n), buffer size (m), and input load (q). Numerical results are presented in various plots with selected settings of these parameters. We assume that enqueueing speed of output buffers is $min(a, m)$ times that of link transmission speed. In theory his assumption gives output-buffered switching an edge over input-buffered switching; however, it does not take into account the hardware cost in practice. We also perform a number of simulations of cut-through MINs with input buffers and output buffers. These simulations serve to check the numerical results produced by our analytical model. Results are compared to ensure the accuracy of the performance analysis. We further investigate the performance of cut-through MINs under bursty traffic and hot-spot traffic. This is done through simulation with various traffic patterns.

4.1 Comparisons of Analytical Results

We employ a time-progression method to calculate the analytical results for cut-hrough switching using input buffering and output buffering. This method works with Eq.(1)-(22) in the following manner. First, we plug some initial values into the performance probabilities of the equations. These include $p_j(k, 0)$, $q_b(k, 0)$, $q_c(k, 0)$, $r_b(k, 0)$, and $r_c(k, 0)$ for stages k, $1 \le k \le n$. From here we compute values of these probabilities at the next time step, namely $p_j(k, 1)$, $q_b(k, 1)$, $q_c(k, 1)$, $r_b(k, 1)$, and $r_c(k, 1)$. The computation goes on until $p_j(k, t)$, $q_b(k, t)$, $q_c(k, t)$, $r_b(k, t)$, and $r_c(k, t)$ converge to some steady-state values. The method is quite efficient, and usually takes about 10 steps to converge. Then we plug these values into the closed forms of the two performance measures and compute the numerical results. Note that using the aforementioned method we can also compute throughput and delay for store-and-forward switching. This is done simply by turning off q_c and r_c in Eqs. (1)-(22).

The following discussion focuses on characterizing performance of cut-through switching in various working conditions. Representative cases are chosen for plots. They allow us to gain insight into performance differences of input-buffered switching versus output-buffered switching, and store-and-forward switching versus cut-through switching.

In the case of shallow buffers, e.g. $m = 1$, cut-through switching yields higher throughput and lower delay than store-and-forward switching. Fig. 5 shows the plots of normalized throughput and delay versus input load using cut-through switching and store-and-forward switching, assuming $a = 2$, $n = 10$ and $m = 1$. We observe that by a wide margin cut-through switching outperforms store-and-forward switching in the two performance measures. Not only can cut-through switching achieve higher throughput than store-and-forward switching, but also it results in lower delay. Between the two-buffered MINs, the input-buffered MIN benefits from cut-through switching more than the output-buffered MIN. For instance, at input load 1.0 cut-through switching boosts the input-buffered MIN on normalized throughput from 0.45 to 0.79, and at the same time reduces normalized delay from 1.55 to 0.78. In this case, input-buffered cut-through switching outperforms output-buffered cut-through switching in both normalized throughput and delay. Because of shallow buffer and heavy input-load, buffers of the output-buffered MIN can easily become full. In an output-buffered switching element, each link is associated solely with an output buffer. Once an output buffer becomes full, no cut can be made through the buffer. Taking advantage of bypass links, however, the input-buffered MIN allows a packet to pass through a full buffer, so long as it does not collide with the head-of-line packet in the buffer.

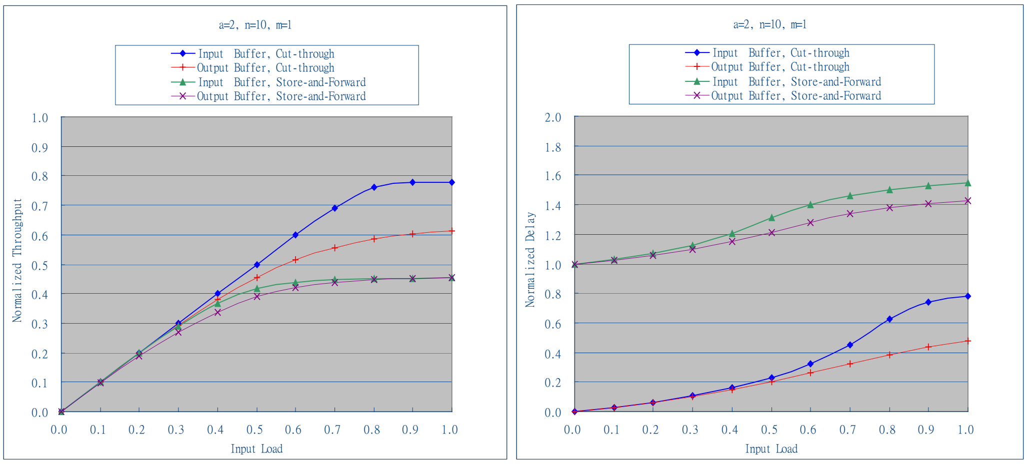

Fig. 5. Normalized throughput and delay versus input load for $a = 2$, $n = 10$ and $m = 1$.

As input load exceeds 0.9, cut-through switching has different effects on input buffers and output buffers. Fig. 6 presents the plots of normalized throughput and delay versus buffer size for cut-through switching and store-and-forward, assuming $a = 4$ and $n = 5$ and input load = 1.0. In the case of input buffers, cut-through switching is helpful in increasing throughput and reducing delay compared with its store-and-forward counter

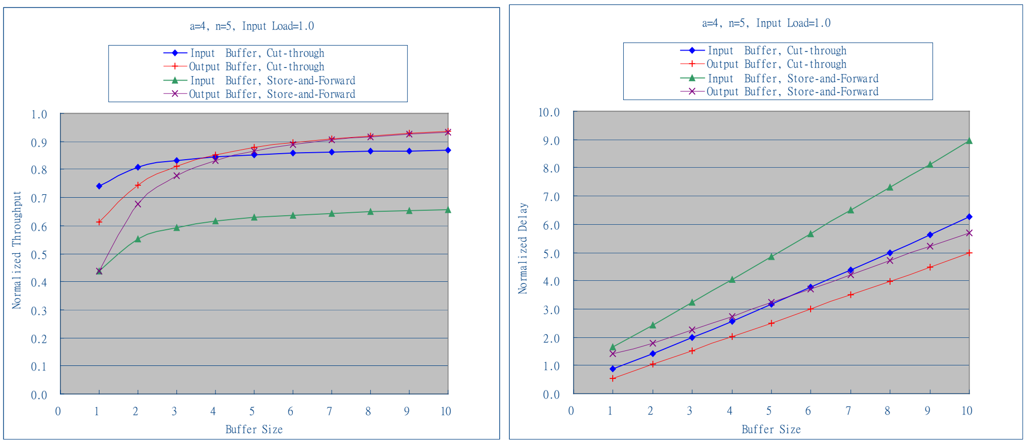

Fig. 6. Normalized throughput and delay versus input load for $a = 4$, $n = 5$ and $q(1) = 1.0$.

part. In the case of output buffers, cut-through switching achieves a significant reduction in normalized delay as buffer size increases. But it is of little help in improving throughput of the MIN with large output buffers. When output-buffered MINs become saturated with packets, enqueueing speed becomes a dominant factor in determining throughput. In this particular example, the maximum enqueueing speed for output buffers is 4. We observe that this is approximately the point where the improvement on throughput due to cut-through switching starts to decline. Because of the advantage of higher enqueueing speed, the output-buffered store-and-forward MIN comfortably outperforms the input-buffered store-and-forward MIN by a wide margin in both throughput and delay. However, it is interesting to note that the margin is narrowed if cut-through switching is employed instead. In some cases of small buffers, the input-buffered cut-through MIN achieves even higher throughput than the output-buffered cut-through MIN.

Figs. 7 and 8 plot normalized throughput and delay versus switch size for the input-buffered $2^{16} \times 2^{16}$ MIN under a heavy input load of 1.0. Switch size varies from 2 to 256, corresponding to 16 stages to 2 stages of the MIN. We observe that the throughput is considerably improved by cut-through switching and that the improvement varies slightly with switch. We also observe that the improvement by cut-through switching is more significant for small switchs than for large switchs. The effect of cut-through switching on normalized delay differs as buffer size varies. The reduction of normalized delay for buffer size 8 is much larger than that for buffer size 1. Nevertheless, the reduction seems insensitive switch size. While the normalized delay for buffer size 1 varies slightly with switch size, the normalized delay for buffer size 8 increases monotonically with switch size.

Now, let us look at Fig. 8. It shows normalized throughput and delay versus switch size for the output-buffered $2^{16} \times 2^{16}$ MIN under a heavy input load of 1.0. The throughput curves show that cut-through switching for large buffers is more effective than for small buffers, especially when switch size is large. The delay curves exhibit a characteristic different from those for input-buffered switching; the difference of normalized delay between store-and-forward switching and cut-through switching narrows as switch size increases. This becomes more apparent for large-sized buffers. As switch size increases,

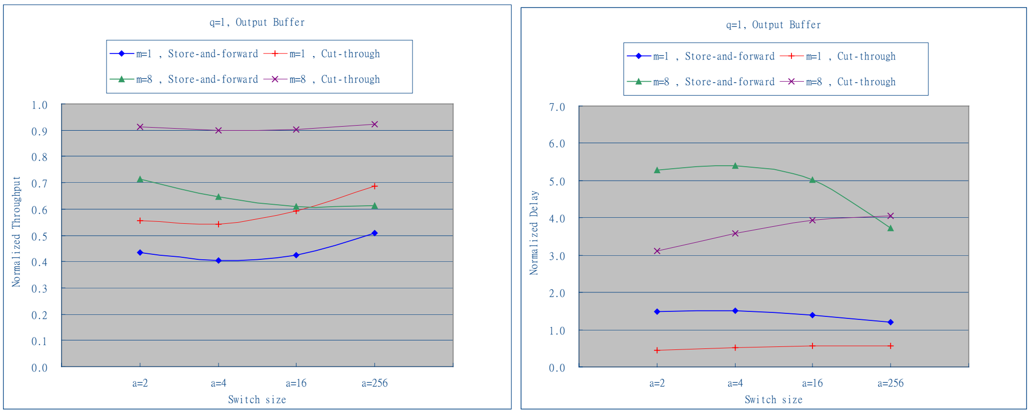

Fig. 7. Normalized throughput and delay versus switch size for the output-buffered $2^{16} \times 2^{16}$ MIN under input load of 1.0.

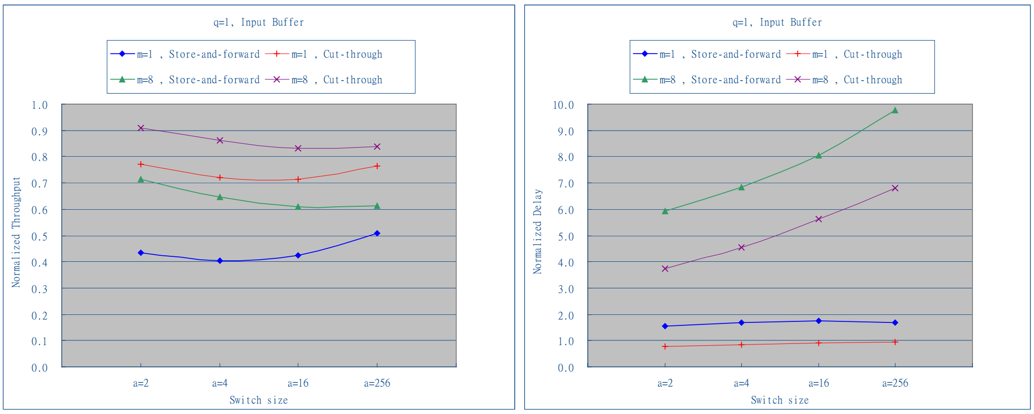

Fig. 8. Normalized throughput and delay versus switch size for the input-buffered $2^{16} \times 2^{16}$ MIN under input load of 1.0.

the number of stages decreases. Actually, the number of buffers in the MIN decreases too. The buffers of large switches are less likely to become empty, and it is less likely to deliver packets by cut-through. On the other hand, in this case, the output-buffered, cut-through MIN operates more like the store-and-forward MIN.

4.2 Simulation Results

We implement a simulator for validating our analysis. The simulator is a program that simulates MINs using cut-through switching with input buffers or output buffers. We assume that a packet transfer from buffer to buffer, including cut-through transfer, takes one cycle to completion. The network inputs are fed with packets generated by a random-number generator at a given load of $q(0)$. Packet destinations are randomized with a uniform distribution. The network is simulated synchronously; all the packets start to transfer at the beginning of a network cycle and complete at the end. Then we advance time to the next cycle, and so on. The simulator is programmable as to switch size, network size, buffer size, and input load. Each simulation runs for a total of 10,000 cycles. Packets are collected at the output links of the last stage. Then throughput and delay are averaged over network size and simulation time for normalization.

Figs. 9 and 10 present two sets of simulation results. They are compared against the corresponding analytical results for a 64 × 64 cut-through MIN using input buffers and output buffers, respectively. Each figure contains two plots of normalized throughput and delay for buffer sizes 1 and 12, with input load varying from 0 to 1.0. The plots indicate that accuracy of the analytical results for throughput degrades as buffer size decreases. In fact, we purposely choose the plots in an attempt to indicate the greatest discrepancy occurring at m = 1. This is the situation where buffers at the first stage are most likely overflow; and many incoming packets, generated by Bernoulli trials, cannot enter the overflowed MINs. This alters the assumed geometric-distribution inter-arrival time of packets. The more packets that are dropped, the higher the variance of inter-arrival time, and the greater the discrepancy. This affects accuracy of our analytical model in predicting normalized throughput of cut-through MINs with small buffers. But the effect of packet loss on accuracy of normalized delay is the opposite.

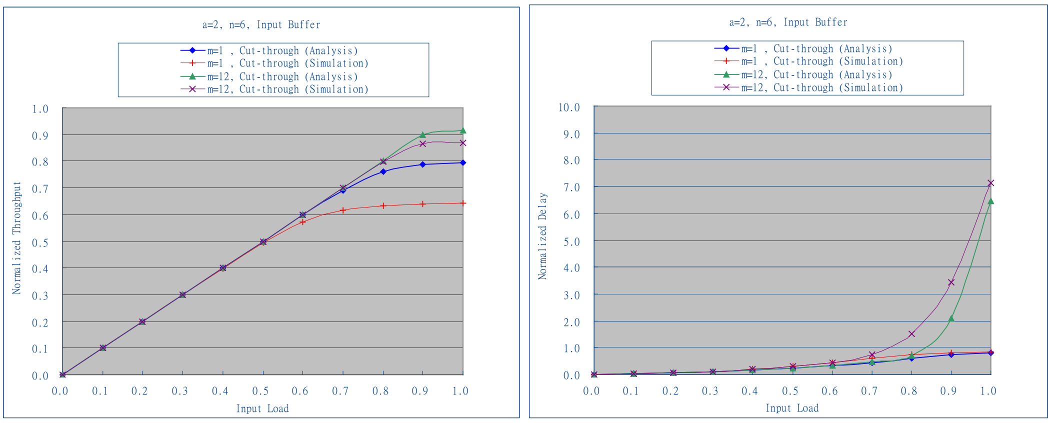

Fig. 9. Comparison between analytical and simulation results for input-buffered MIN with $a = 2$ and $n = 6$.

When buffer size increases, more packets are brought into the switch. Packets then are prone to experience drastic changes in queueing time as they proceed from stage to stage. On the other hand, this causes higher variance in queueing time in the case of large buffers. Therefore, disagreement of normalized delay between the analytical and simulation results becomes apparent as buffer size increases.

4.3 Performance Under Bursty Traffic and Hot-Spot Traffic

We also study the effect of cut-through switching on the buffered MIN under two different types of traffic: bursty traffic and hot-spot traffic. The study is conducted through simulations using the aforementioned system parameters. For bursty traffic, we

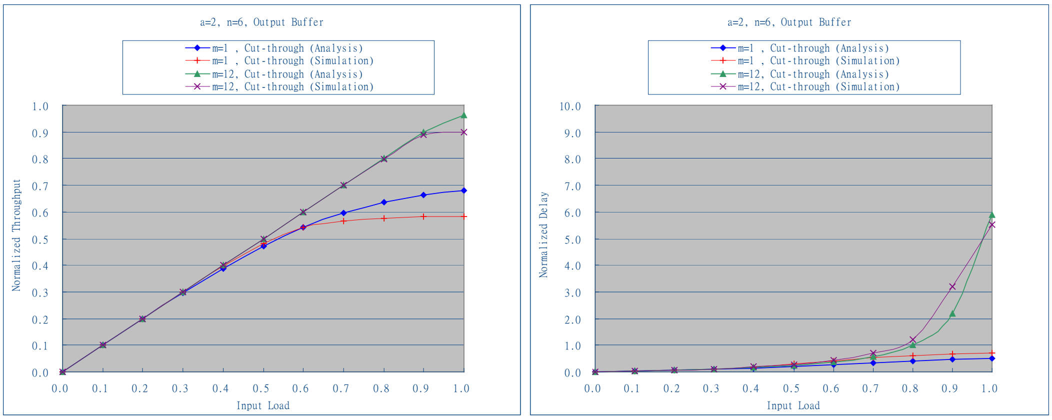

Fig. 10. Comparison between analytical and simulation results for output-buffered Min with $a = 2$ and $n = 6$.

assume a commonly-used on-off bursty source to produce packets entering the first stage of the buffered MIN. The bursty source, which is a two-state machine, alternates between active and idle periods with two independent geometric distributions. Packets arrive in consecutive time slots for the duration of an active period with at least one packet. An idle period, however, can be empty. Fig. 11 shows two plots of normalized throughput with /under a bursty traffic of length 20 on the input-buffered and output-buffered MINs with $a = 2$ and $n = 6$. Compared with store-and-forward switching, cut-through switching yields higher throughput for both input-buffered MINs and output-buffered MINs with single buffers. In the case of $m = 12$, the effectiveness of cut-through switching remains significant for input-buffered MINs, but it is of little significant for output-buffered MINs.

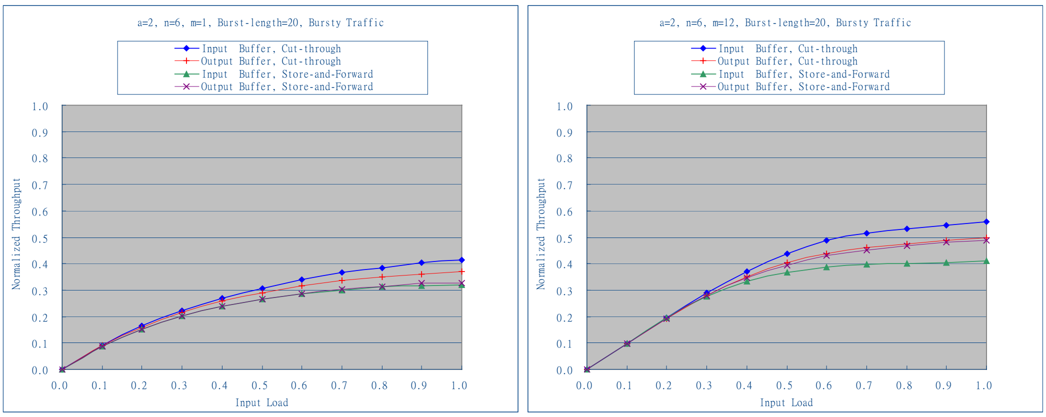

Fig. 11. Normalized throughput versus input load for input-buffered and output-buffered MINs with $a = 2$ and $n = 6$ under bursty traffic of length 20.

We consider a hot-spot traffic model in the simulation study for cut-though switching under nonuniform traffic [13]. The hot-spot traffic model assumes that one of the network outputs, called hot-spot output, receives more packets than the remaining network outputs, and that packets are destined the remaining outputs with equal probability. Distribution of destination address is determined by a hot-spot ratio, h, which is related to input load, p, by equation

$$p = ph + p(1 - h).$$

In the expression, ph represents the probability that a network input in a time slot has a packet destined to the hot-spot output, and $p(1 - h)$ is the probability a network input in a time slot has a packet destined to the outputs with uniform distribution. As a consequence, on average, the hot-spot output becomes the destination of $phN + p(1 - h)$ packets where N is the number of network inputs. Fig. 12 gives a representative example for illustrating the throughput performance of cut-through switching under hot-spot traffic with $h = 0.1$. When buffer size equals one, cut-through switching substantially improves throughput for input-buffered MINs and output-buffered MINs as well. When buffer size equals twelve, throughput for cut-through input-buffered MINs is 0.4299. This is approximately three times higher than throughtput for store-and-forward MINs. However, in the case of output-buffered MINs, cut-through switching makes virtually no improvement on throughput. This indicates that cut-through switching is more effective for input buffered MINs than for output-buffered MINs.

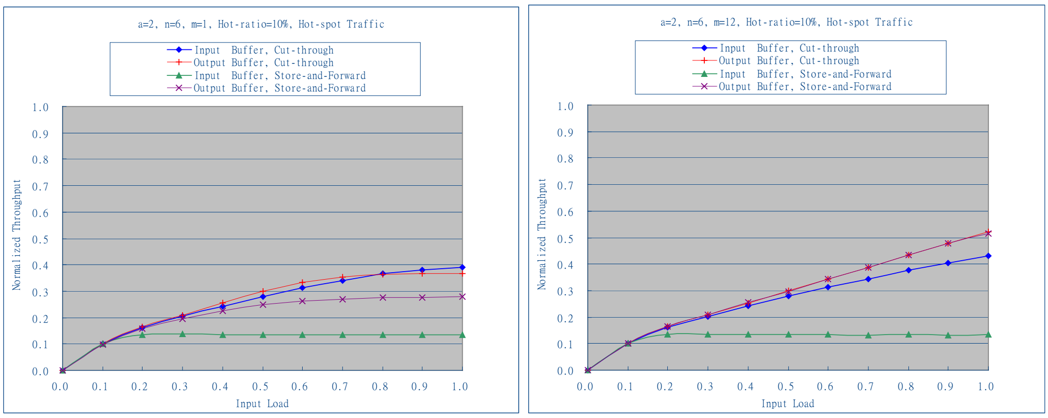

Fig. 12. Normalized throughput versus input load for input-buffered and output-buffered MINs with $a = 2$ and $n = 6$ under hot-spot traffic of length 20.

5. CONCLUSIONS

We have presented a new analytical model for evaluating cut-through switching on buffered MINs. Two classes of buffered MINs were considered: input-buffered MIN and output-buffered MIN. Our analytical model is a general one in the sense that it assumes arbitrary switch size, buffer size, and network size. Because of this generality, the proposed model is able to analytically exhibit several essential properties which were unknown before, affecting the performance of cut-through switching on buffered MINs. Of particular interest is the operating condition under which cut-through switching can make significant performance improvement over store-and-forward switching.

We found that cut-through switching works extremely well when switch size and switch buffer are small. It simultaneously increases throughput and reduces delay for both input-buffered MINs and output-buffered MINs. But as switch size increases, the effectiveness of cut-through switching on input-buffered and output-buffered MINs diverge. With the aid of cut-through switching, input-buffered MINs with large switches enjoy a great gain in normalized throughput and simultaneously a substantial reduction in normalized delay. As switch size increases, cut-through switching effectively reduces the delay for the output-buffered MIN. However, it offers little help in boosting the throughput for the output-buffered MIN. Especially when buffer size is greater than five, cut-through switching is of virtually no help to the output-buffered MIN. In the case of store-and-forward switching, the output-buffered MIN outperforms the input-buffered MIN in throughput by a wide margin. But the margin is reduced when cut-through switching is employed. Finally, by means of simulation we evaluate the performance of cut-through switching on buffered MINs under bursty and hot-spot traffic. Results show that the input-buffered MIN benefits more from cut-through switching than the output-buffered MIN. Cut-through switching helps the input-buffered MIN improve the throughput performance in response to bursty and hot-spot traffic. However, the improvement is considerably limited for output-buffered MINs, particularly for MINs with large output buffers.

REFERENCES

1. P. Kermani and L. Kleinrock, “Virtual cut-through: a new computer communication switching technique,” *Computer Networks*, Vol. 3, 1979, pp. 267-286.
2. C. P. Kruskal and M. Snir, “The performance of multistage interconnection networks for multiprocessors,” *IEEE Transactions on Computers*, Vol. 32, 1983, pp. 1091-1098.
3. Y. C. Jeng, “Performance analysis of a packet switch based on single-buffered banyan network,” *IEEE Journal Selected Areas Communications*, Vol. SAC-3, 1983, pp. 1014-1021.
4. T. Szymanski and S. Shaikh, “Markov chain analysis of packet switched banyans with arbitrary switches, queue sizes, link multiplicities and speedups,” in *Proceedings of IEEE INFOCOM 1989*, 1989, pp. 960-971.
5. H. Yoon et al., “Performance analysis of multibuffered packet-switching networks in multiprocessor systems,” *IEEE Transactions on Computers*, Vol. 39, 1990, pp. 319-327.
6. R. G. Bubenik and J. S. Turner, “Performance of a broadcast packet switch,” *IEEE Transactions on Communications*, Vol. 37, 1989, pp. 60-69.
7. I. Widjaja et al., “ The effect of cut-through switching on the performance of buffered banyan networks,” *Computer Networks and ISDN Systems*, Vol. 26, 1993, pp. 139-159.
8. Y. Pan and R. Melhem, “Short circuits in buffered multi-stage interconnection networks,” *Compt. Journal*, Vol. 33, 1990, pp. 323-329.

9. A. Abo-Taleb and H. T. Mouftah, " Delay analysis under a general cut-through technique in computer networks," *IEEE Transactions on Communications*, Vol. 35, 1987, pp. 356-359.
10. J. W. Dolter et al., " Performance analysis of virtual cut-through switching in HARTS: a hexagonal mesh computer," *IEEE Transactions on Computers*, Vol. 40, 1991, pp. 669-680.
11. C. L. Wu and T. Y. Feng, "On a class of multistage interconnection networks," *IEEE Transactions on Computers*, Vol. 29, 1980, pp. 694-702.
12. J. H. Patel, "Performance of processor-memory interconnections for multiprocessors," *IEEE Transactions on Computers*, Vol. 30, 1981, pp. 771-780.
13. G. F. Pfister and V. A. Norton, "'Hot-spot contention and combining in multistage interconnection networks," *IEEE Transactions on Computers*, Vol. 34, 1985, pp. 943-948.
14. L. Tassiulas, "Cut-through switching, pipelining and scheduling for network evaluation," *IEEE/ACM Transactions on Networking*, Vol. 7, 1999, pp. 88-97.
15. W. Qiao, L. M. Ni, and T. Rokicki, "Adaptive-trail routing and performance evaluation in irregular networks using cut-through switches," *IEEE Transactions on Parallel and Distributed Systems,* Vol. 10, 1999, pp.1138-1158.
16. D. A. Schupke and A. Riedl, "Packet transfer delay comparison of a store-and-forward and a cut-through resilient packet ring," *International Zurich Seminar on Access, Transmission, Communications*, 2002, pp. 12-1-12-5.

Chan L. Liao (廖建良) received the B.S. degree in information and computer education from National Taiwan Normal University in 1991, and the M.S. degree in computer science from National Chung-Hsing University. He is currently a lecture of National Taichung Home Economical and Commercial Senior Vocational School in Taichung, Taiwan. His interests include performance analysis, computer networks, and computer architecture.

Woei Lin (林偉) received the B.S. degree from National Chiao-Tung University, Taiwan, in 1978, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Texas at Austin, USA in 1982 and 1985, respectively.

He is currently a Professor with the Institute of Computer Science, National Chung-Hsing University, Taiwan, R.O.C. He was an Associate Professor of Department of Electrical Engineering at University of Hawaii, and an Assistant Professor of Department of Electrical Engineering at Penn State University, USA. His research areas include high-speed switching, computer networks, parallel processing, and performance evaluation.