Performance Analysis of the Multiple Input-Queued Packet Switch With the Restricted Rule

Hakyong Kim, *Associate Member, IEEE,* and Kiseon Kim, *Senior Member, IEEE*

***Abstract*—The multiple input-queued (MIQ) switch is the switch which manages multiple (m) queues in each input port, each of which is dedicated to a group of output ports. Since each input port can switch up to m cells in a time slot, one from each queue, it hardly suffers from the head-of-line (HOL) blocking which is known as the decisive factor limiting the throughput of the single input-queued (SIQ) switch. As the result, the MIQ switch guarantees enhanced performance characteristics as the number of queues m in an input increases. However, the service of multiple cells from an input could cause the internal speedup or expansion of the switch fabric, diluting the merit of high-speed operation in the conventional SIQ scheme. The restricted rule is contrived to circumvent this side effect by regulating the number of cells switched from an input port. In this paper, we analyze the performance of the MIQ switch employing the restricted rule. For the switch using the restricted rule, the closed formulas for the throughput bound, the mean cell delay and average queue length, and the cell loss bound of the switch are derived as the function of m, by generalizing the analysis for the SIQ switch by Hui *et al.*, 1987.**

***Index Terms*—Multiple input queueing (MIQ), restricted rule, free rule.**

NOMENCLATURE

N	Switch size.
m	Number of FIFO queues in an input port.
T	Average throughput.
T_i	Throughput for output group i.
$T_{i,\text{sat}}$	Saturation throughput for output group i.
$\epsilon(x)$	Indication function.
N_j	Number of HOL requests for the same output port j.
N_j'	N_j for the next time slot.
A_j	Number of fresh HOL arrivals for output port j.
$N^{(b)}$	Number of all HOL cells that became blocked at the end of a time slot.
λ	Mean arrival rate for an input.
μ	Mean service rate of an input.
ρ	Steady-state probability that a queue has a fresh HOL cell which is just moved to the HOL position, given that the queue is not blocked during the previous slot.
λ_i	Effective arrival rate for output group i.
μ_i	Effective service rate of output group i.
ρ_i	Effective value of ρ for output group i.
R_i	Number of unblocked HOL cells of the queues attending the arbitration for output group i.
S_i	Number of input ports selected in the arbitration for output group i.
δ_i	Sum of the throughputs for the first i output groups.
λ_{avg}	Average of λ_i's for $1 \le i \le m$.
μ_{avg}	Average of μ_i's for $1 \le i \le m$.
ρ_{avg}	Average of ρ_i's for $1 \le i \le m$.
K	Number of cells in a queue just before the arbitration phase.
p_k	Steady-state probability distribution of K.
$G(z)$	Probability generating function for p_k.
$\bar{K}$	Average queue length.
D	Mean cell delay time.

Manuscript received August 12, 1999; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor H. S. Kim.

H. Kim is with the Research and Development Center, Corecess Inc., Seoul 135-280, Korea (e-mail: hykim@ieee.org).

K. Kim is with Department of Information and Communications, Kwang-Ju Institute of Science and Technology, Kwang-Ju 500-712, Korea (e-mail: kskim@kjist.ac.kr).

Publisher Item Identifier 10.1109/TNET.2003.813015.

I. INTRODUCTION

IT IS a well-known fact that the throughput of the single input-queued (SIQ) packet/cell switch is limited to 58.6% for an independently and identically distributed Bernoulli arrival traffic with the cell destinations uniformly distributed over all output ports, namely, the homogeneous Bernoulli arrival traffic, when the switch size is infinite [1], [2]. In order to overcome the throughput limit of the SIQ switch, many alternative queueing methods and/or scheduling schemes are conceived, such as window policy, input smoothing, and the multiple input-queueing schemes [2]–[6]. The essence of these schemes is to allow one of the cells behind the head-of-line (HOL) cell to be switched to an idle output when the HOL cell is blocked, or to allow multiple cells to be the HOL cell and hence increase the opportunity for an input port to serve cells. Among those schemes, the multiple input-queueing scheme is drawing much attention recently in that it can provide both high-speed operation and high switching performance.

The multiple input-queued (MIQ) switch is the switch equipped with multiple queues in every input. Since the queues are independent logically or physically of other queues in the same input, an input can switch plural cells in a time slot, which leads to the internal speedup or to the switch fabric expansion. The restricted rule has been developed to resolve such problems in the MIQ switch. The rule regulates an input to switch at most one cell in a time slot. Restricting the number of cells to be switched, in a sense, implies that the maximum achievable performance may be degraded a little bit, but makes it feasible to develop the high-speed and high-performance switch.

In connection with the restricted rule, many scheduling algorithms have been suggested [43]–[52], and some of them compensate for the performance degradation caused by the restricted rule by finding maximum matches between input and output ports. Another cardinal issue related to the restricted rule is the implementation and time complexity of the scheduling algorithm. In 1993, Anderson *et al.* proposed a parallel scheduling algorithm called the parallel iterative matching (PIM) algorithm [43]. In this algorithm, packets are scheduled in each time slot through a sequence of three actions of *request*, *grant*, and *accept*, making the algorithm simple and easily realizable in hardware. Most scheduling algorithms for the MIQ switch designed after Anderson's work are essentially based on the philosophy of the PIM, that is, three-phase operation, and try to find maximum matches between input and output ports.

In this paper, we will focus on the performance side of the MIQ switch rather than other issues, since there is a limited number of studies on the performance analysis for the MIQ switch, and most of the studies apply only to a particular MIQ switch called the virtual output-queued switch. We first develop the queueing model for the MIQ switch employing the restricted rule and then analyze the performance measures of the switch such as saturation throughput, mean cell delay and average queue length, and cell loss probability, by generalizing the analysis results for the SIQ packet switch [1].

This paper is organized as follows. In Section II, we investigate the concept of multiple input-queueing and two types of cell selection rules: the restricted rule and the free rule. In Sections III and IV, performance of the MIQ switch is analyzed in terms of the number of queues in an input port and the offered load. In deriving performance measures of the MIQ switch, we use two types of queueing models, one for the throughput bound and the other for the mean cell delay and the cell loss probability. Finally, we conclude in Section V with a discussion on the application of the switch in the future broadband communications systems.

II. MIQ Packet Switch

Compared to the output-queued switch, the (single) input-queued switch has an inherent merit: it operates at the same speed as the external link speed, making the input-queued switch more suitable for the high-speed switching system. Moreover, the input queueing is considered easier to implement in hardware than its output counterpart. On the other hand, the input-queued switch suffers from the HOL blocking so that its maximum attainable throughput is limited to 0.586 for homogeneous traffic [1], [2]. Therefore, many buffering and scheduling schemes have been engineered to overcome the throughput limit of the SIQ switch. They fall into five large categories:

1) examining the first k packets in a first-in-first-out (FIFO) queue instead of the HOL packet only; this scheme is called *window policy*, or sometimes called the look-ahead scheme or bypass scheme [6]–[18];
2) expanding the internal switch bandwidth either by enlarging the switch dimension or by replicating the switch fabric; this scheme is called *input smoothing* or *input expanding* [2]–[7], [18]–[21];
3) employing multiple FIFO queues instead of a single FIFO in every input port; this scheme is referred to as the *multiple input-queued (MIQ) scheme* or *bifurcated input-queued scheme* [22]–[37];
4) input grouping or trunking [22], [38]–[40];
5) dropping or discarding the blocked HOL cells in the FIFO queue [2], [18].

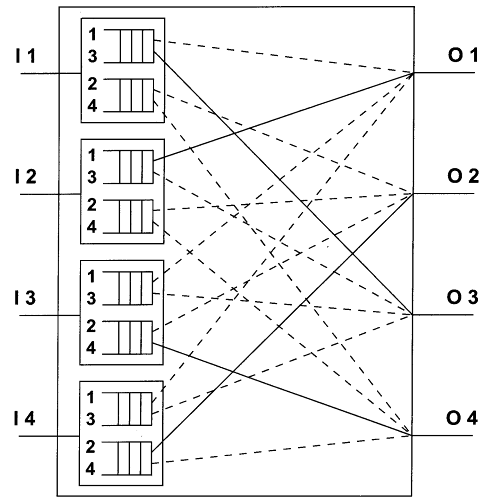

Fig. 1. Multiple input-queued switch.

Among them, it is reported that the MIQ scheme can dramatically improve the switching performance and, thus, lift the throughput bound of the SIQ switch up to 1.0 when the number of queues in an input is equal to or larger than the switch size [30]–[37].

The MIQ switch, as shown in Fig. 1, deploys multiple (m) queues in each input port, each of which is dedicated to a group of output ports. When m is equal to one, that is, the single queue is dedicated to all output ports, the MIQ switch corresponds to the conventional SIQ switch. When m is equal to the switch size N, that is, each queue is dedicated to an output port, the MIQ switch is referred to as the virtual output-queued (VOQ) switch. In the VOQ switch, the probability that HOL blocking occurs becomes zero [42]. As we will see later, the HOL blocking probability decreases as the number of queues (m) in an input increases.

The value of m is usually taken to be an exponential of $2(1, 2, 4, 8, \ldots)$ in the range from 1 to N, and N is assumed to be the multiple of m. (The range of m can be larger than N, which makes the switch not practical.) Therefore, each queue stores cells for N/m output ports exclusively and exhaustively. That is, output groups do not overlap with others in terms of output ports. We can here make an important observation that the switch can be split into m identical subswitches whose sizes are $N \times N/m$ and each subswitch will correspond to an output group. (The words *subswitch* and *output group* will be used interchangeably in this paper.) This observation is the starting point of the analysis in the following section.

The MIQ switch can switch up to m cells from an input port in a time slot since an input port manages m physically or logically distinct queues. However, the total number of cells switched from all inputs must not exceed N and each output can receive at most one cell in every time slot. Even though the service of multiple cells from an input port is an essential reason for the performance enhancement in the MIQ switch, it gives rise to an internal speedup (or an expansion in the switch fabric) at the same time, diluting the salient merit of the input-queued switch. The internal speedup makes the switch unsuitable for the high-speed and high-performance switch and, thus, is one of the important issues to be resolved for future switching systems.

Therefore, most scheduling schemes developed for the MIQ switch assume that each input can switch at most one cell in a time slot [43]–[52]. The rule governing this assumption is referred to as the *restricted rule* [35], [37]. Under the restricted rule, the switch operates at the same speed as the external link speed. On the other hand, the *free rule* allows an input port to switch up to m cells in a time slot with the potential internal speedup provided that the total number of switched cells is no more than N. In [37], the authors showed that the saturation throughput of the MIQ switch converges to 1.0 as m increases, irrespective of the rules employed. It also shows that the free rule yields slightly higher saturation throughput for moderate values of m.

The performance for the free rule has been analytically evaluated in different ways [30]–[41]. In [30], the authors derived the closed expression for the saturation throughput of the MIQ switch using the free rule in terms of the number of queues in an input. Performance of the restricted rule, however, has usually been evaluated through computer simulation. This is because input ports and their queues are mutually dependent on one another in regulating the number of cells switched from each input, making the queueing analysis difficult. The performance analysis for the restricted rule was found in [32] and [56] in terms of the saturation throughput. In the following two sections, we analyze the performance of the MIQ switch employing the restricted rule. In Section III, the throughput of the MIQ switch under the restricted rule is analyzed after developing a queueing model for a specific queue in an input port. In Section IV, mean cell delay and cell loss probability are derived after developing a queueing model with average parameters.

III. Saturation Throughput

In this section, we first develop a queueing model for an MIQ switch using the restricted rule and then analyze it in terms of the saturation throughput.

A. Queueing Model for the Throughput

For the MIQ switch, which we will analyze in following sections, we assume that the switch size is $N \times N$ and the number of FIFO queues in an input port is m $(1 \leq m \leq N)$. The interconnection network of the switch is internally nonblocking. Each queue is assumed to store cells for an output group including N/m output ports. For the switch, we assume further that time is slotted and the slots carry cells or fixed-length packets. Cells arrive at input ports or queues just after the time boundaries, and depart from the queues just before the time boundaries. This is known as the early-arrival model [53]. Traffic arriving at each input is assumed to be independent and identical to one another. The arrived traffic is assumed to be uniformly distributed for N outputs or, namely, uniformly distributed for m queues in an input port. We refer to this kind of traffic as the *homogeneous Bernoulli arrival traffic*. During a time slot, each input can switch no more than one cell and each output can receive at most one cell.

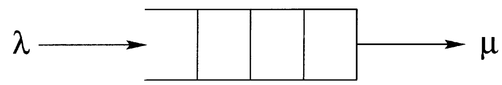

Fig. 2. Queueing model for an input of the MIQ switch.

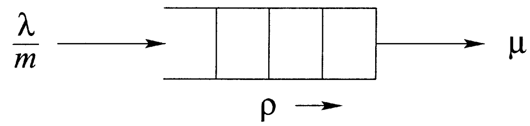

Fig. 3. Queueing model for a queue.

For the MIQ switch, let the mean arrival rate and the mean service rate for an input port be λ and μ, respectively, as shown in Fig. 2. In an input port, then, the mean arrival rate for a queue becomes λ/m since the arrived traffic is distributed uniformly between m queues. On the other hand, the mean service rate for a queue is still μ. Therefore, the resulting queueing model for a queue can be represented as shown in Fig. 3. In the figure, let ρ be the steady-state probability that a queue has a fresh HOL cell which is just moved to the HOL position, given that the queue is not blocked during the previous slot. The fresh HOL cell means the cells that move up to the HOL position. They were either queued or, if the queue was empty, they are new.

B. Saturation Throughput for the Restricted Rule

As mentioned before, arbitrations are assumed to be performed sequentially for each output group or subswitch and the input ports selected in earlier arbitrations should be excluded in later arbitrations for remaining output groups or subswitches in some ways. Differently from the MIQ switch using the free rule, the throughput is different from subswitch to subswitch in the MIQ switch using the restricted rule. This is because arbitrations performed later are dependent on previous arbitration results in the same time slot [35]. Therefore, the throughput for the entire switch can be obtained by taking expectation on the sum of the throughputs for m output groups or subswitches.

We number output groups from 1 to m where the output group including the first N/m output ports is referred to as output group 1. For convenience, let us assume that the arbitration starts from output group 1. Letting T_1 designate the throughput for the first $N \times N/m$ subswitch corresponding to output group 1, then T_1 becomes

$$T_1 = \frac{m}{N} \sum_{j=1}^{\frac{N}{m}} E[\epsilon(N_j)] = E[\epsilon(N_j)] \tag{1}$$

where the indication function $\epsilon(x)$ is

$$\epsilon(x) = \begin{cases} 1, & x \geq 1 \\ 0, & x = 0 \end{cases}$$

and N_j is the number of HOL requests for the same output port j. As long as m is not equal to N, the MIQ switch suffers from HOL blocking as mentioned in Section II and the state of HOL blocking is characterized by N_j. Note here that the index j for output ports in (1) takes its value between 1 and N/m since the arbitration for output group 1 concerns the first N/m output ports in output group 1. The number of all HOL cells that become blocked at the end of a slot, $N^{(b)}$, is

$$N^{(b)} = \sum_{j=1}^{\frac{N}{m}} N_j - \sum_{j=1}^{\frac{N}{m}} \epsilon(N_j). \tag{2}$$

Taking expectation on both sides of (2) and combining it with (1), we get

$$T_1 = E[N_j] - \frac{m}{N} E[N^{(b)}]. \tag{3}$$

We can get T_1 by computing the last two terms in (3) in terms of λ_1, since we have $T_1 = \lambda_1$ in steady state. Note that λ_1 is the mean or effective arrival rate for output group 1.

Two terms in the right-hand side of (3) can be obtained as in Appendices A and B. Substituting (A.4) and (B.3) into (3), we have the throughput for the first N/m output ports (or for output group 1):

$$T_1 = \lambda_1 + \frac{\lambda_1^2}{2(1-\lambda_1)} - \frac{m}{N}\left(N - \frac{N}{m}\frac{\lambda_1}{\rho}\right)\Bigg|_{\lambda_1 = T_1} \tag{4}$$

from which we obtain

$$(2-\rho)T_1^2 - 2(m\rho+1)T_1 + 2m\rho = 0 \tag{5}$$

or

$$T_1 = \frac{m\rho + 1 - \sqrt{(m\rho+1)^2 - 2m\rho(2-\rho)}}{(2-\rho)}. \tag{6}$$

The saturation throughput for output group 1 is obtained by setting $\rho = 1$ in (5), and

$$T_{1,\text{sat}} = m + 1 - \sqrt{m^2+1}.$$

Letting S_1 designate the number of input ports selected in the arbitration for output group 1, it becomes

$$S_1 = \frac{N}{m} T_1.$$

Now, let us turn our attention to output group 2. The number of output ports in output group 2 is still N/m and, hence, (1) through (3) can be used without any modifications for output group 2 except for the suffix. In the arbitration for output group 2, however, the number of unblocked queues, R_2, changes into

$$R_2 = (N - S_1) - N^{(b)} \tag{7}$$

since we prevent S_1 input ports, selected in the arbitration for output group 1, from attending the arbitration for output group 2. That is, the term $(N - S_1)$ of (7) designates the number of input ports (or queues) which are eligible for attending the arbitration for output group 2. Taking expectation on both sides of (7) and using the flow conservation rule of $E[R_2]\rho = (N/m)\lambda$, we have

$$\begin{aligned} E[N^{(b)}] &= N - E[S_1] - \frac{N}{m}\frac{\lambda_2}{\rho} \\ &= N - \frac{N}{m}\delta_1 - \frac{N}{m}\frac{\lambda_2}{\rho} \end{aligned} \tag{8}$$

where $\delta_1 = \lambda_1$. Substituting (A.4) and (8) into (3), we get

$$(2-\rho)T_2^2 - 2((m-\delta_1)\rho + 1)T_2 + 2(m-\delta_1)\rho = 0. \tag{9}$$

The throughput for output group 2, T_2, is obtained by solving (9). Setting $\rho = 1$ in (9), we immediately have the saturation throughput for output group 2:

$$T_{2,\text{sat}} = m - \delta_1 + 1 - \sqrt{(m-\delta_1)^2 + 1}.$$

If we assume that m is equal to two, that is, if each input manages only two queues as in the Odd-Even switch [54], we have to stop here. Then, the average saturation throughput for the MIQ switch with $m = 2$ is obtained by the arithmetic mean. That is

$$T = \frac{T_1 + T_2}{2}.$$

The same reasoning can be used for output group 3 and later until output group m when m is greater than or equal to 3. Then, the number of unblocked queues becomes

$$R_i = \left(N - \sum_{j=1}^{i-1} S_j\right) - N^{(b)}$$

where

$$S_j = \frac{N}{m} T_j.$$

When the offered load to an input port is ρ, the throughput for output group i is

$$(2-\rho)T_i^2 - 2((m-\delta_{i-1})\rho + 1)T_i + 2(m-\delta_{i-1})\rho = 0 \tag{10}$$

or

$$T_i = \frac{(m-\delta_{i-1})\rho + 1 - \text{SQRT}_i}{2-\rho} \tag{11}$$

where

$$\text{SQRT}_i = \sqrt{((m-\delta_{i-1})\rho+1)^2 - 2\rho(2-\rho)(m-\delta_{i-1})}$$

and

$$\delta_i = \begin{cases} 0, & i = 0 \\ \sum_{j=1}^{i} T_j, & i \geq 1. \end{cases}$$

The saturation throughput for output group i is expressed by setting $\rho = 1$ as

$$T_{i,\text{sat}} = m - \delta_{i-1} + 1 - \sqrt{(m-\delta_{i-1})^2 + 1}.$$

Consequently, the average throughput of the MIQ switch using the restricted rule is given by

$$T = \frac{1}{m}\sum_{i=1}^{m} T_i = \frac{1}{m}\delta_m. \tag{12}$$

The total saturation throughput is also given by (12) with setting $\rho = 1$.

Fig. 4 plots the saturation throughput of the MIQ switch employing the restricted rule for different values of m. The figure shows that the saturation throughput converges at 1.0 as the number of queues m increases. Note that the saturation

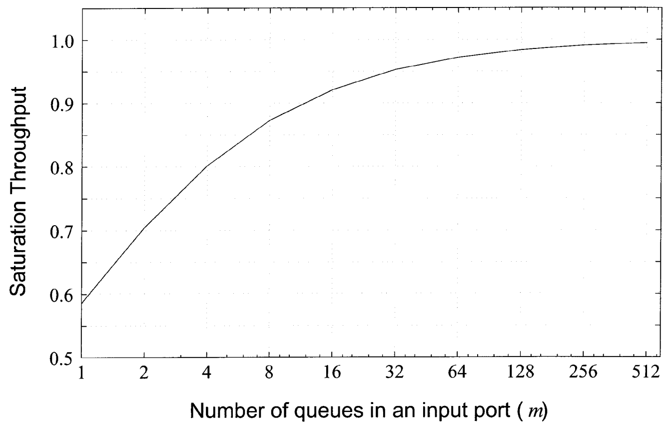

Fig. 4. Saturation throughput of the MIQ switch employing the restricted rule.

TABLE I
SATURATION THROUGHPUT $T_{\rm sat}$ AND δ_m FOR DIFFERENT VALUES OF m. λ IS OBTAINED BY DIVIDING δ_m WITH m. $T_{\rm free,sat}$ IS THE SATURATION THROUGHPUT FOR THE FREE RULE

m	T_{sat}	δ_m	$T_{free,sat}$
1	0.586	0.586	0.586
2	0.705	1.410	0.764
4	0.802	3.206	0.877
8	0.873	6.981	0.938
16	0.921	14.738	0.969
32	0.953	30.482	0.984
64	0.972	62.215	0.992
128	0.984	125.939	0.996
256	0.991	253.656	0.998
512	0.995	509.367	0.999

throughput for the restricted rule is slightly lower than that for the free rule since each input is regulated to serve at most one cell under the restricted rule [37]. Table I shows the saturation throughput for different values of m, as well as δ_m.

Considering the MIQ switch under the free rule, all inputs can attend all arbitration rounds in the same time slot even though they are selected once or more in previous arbitrations. This implies that the throughput for every arbitration is equal to that for the first arbitration of the restricted rule. Then, the throughput of the MIQ switch using the free rule is equal to (6) except the suffix

$$T_{\rm free} = \frac{m\rho + 1 - \sqrt{(m\rho+1)^2 - 2m\rho(2-\rho)}}{2-\rho}. \tag{13}$$

The saturation throughput is obtained by setting $\rho = 1$ in (13):

$$T_{\rm free,sat} = m + 1 - \sqrt{m^2+1} \tag{14}$$

and is tabulated in Table I together with the saturation throughput for the restricted rule. The expressions of (13) and (14) are also found in other studies on the free rule [29], [31], [37], [56].

IV. Mean Cell Delay Time and Cell Loss Probability

In this section, we focus our attention on the mean cell delay and the cell loss probability of the MIQ switch employing the restricted rule. Unlike the average throughput obtained in the previous section, the mean cell delay and the cell loss probability cannot be evaluated by the arithmetical mean since both the (average) mean cell delay and the (average) cell loss probability are dominated by the worst mean cell delay and the worst cell loss probability, respectively. This means that the queueing model developed in Section III-A becomes useless, and as the result, we have to develop another queueing model for the mean cell delay and the cell loss probability. Fortunately, we can evaluate the mean cell delay and the cell loss probability in terms of the average throughput obtained in the previous section. In the following sections, therefore, we evaluate them after developing another equivalent queueing model in terms of the average parameters.

A. Queueing Model for the Mean Cell Delay and the Cell Loss Probability

Under the restricted rule and the assumption of sequential arbitration in a time slot, parameters describing the queueing behavior of different output groups may change. Hence, we need to quantify the effective queueing parameters for each output group rather than for overall output groups.

Let λ_i, μ_i, and ρ_i designate the effective arrival rate, the effective service rate, and the effective value of ρ for output group i, respectively.

Then the effective service rate for output group i can be expressed as

$$\begin{aligned}\mu_i &= \mu - \frac{1}{m}\sum_{j=0}^{i-1}\mu \\ &= \mu - \frac{\delta_{i-1}}{m}\mu.\end{aligned}$$

Letting $\mu_{\rm avg}$ be the average service rate, it becomes

$$\begin{aligned}\mu_{\rm avg} &= \frac{1}{m}\sum_{i=1}^{m}\mu_i \\ &= \frac{1}{m}\left(m - \frac{1}{m}\sum_{i=1}^{m}\delta_{i-1}\right)\mu \\ &= \frac{f(m)}{m}\mu.\end{aligned} \tag{15}$$

Here we use the definition of $f(m)$ in (C.3). Next, let $\rho_{\rm avg}$ be the average value of ρ_i's. Similarly to the average service rate shown above, it becomes

$$\rho_{\rm avg} = \frac{f(m)}{m}\rho. \tag{16}$$

Note, however, that the effective arrival rate at a queue λ_i is still λ/m regardless of output groups since the arrival traffic is not affected by the restricted rule. Subsequently, the resulting queueing model can be described by the new parameters:

- $\lambda_{\rm avg} = \dfrac{\lambda}{m}$
- $\mu_{\rm avg} = \dfrac{f(m)}{m}\mu$
- $\rho_{\rm avg} = \dfrac{f(m)}{m}\rho$

The corresponding queueing model is shown in Fig. 5.

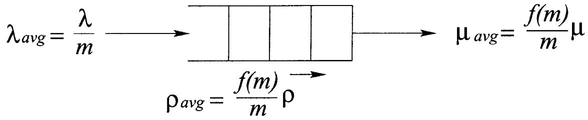

Fig. 5. Queueing model for a queue in terms of average parameters.

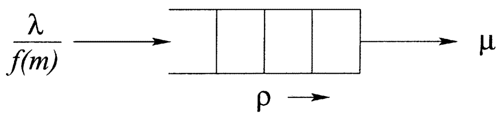

Fig. 6. Equivalent queueing model to the queueing model in Fig. 5.

For the queueing model shown in Fig. 5, the mean number of time slots until a fresh HOL cell is served is $1/\mu_{\rm avg}$ since the number of slots distributes according to the geometric distribution. After a cell is served, the mean number of slots before the arrival of a fresh HOL cell is $(1-\rho_{\rm avg})/\rho_{\rm avg}$ since the number of slots distributes according to the modified geometric distribution. In steady state, the sum of these two terms equals the interarrival time of cells at a queue, namely, m/λ:

$$\frac{1}{\mu_{\rm avg}} + \frac{1-\rho_{\rm avg}}{\rho_{\rm avg}} = \frac{m}{\lambda}. \tag{17}$$

Substituting (15) and (16) into (17), we have

$$\frac{1}{f(m)}\left(\frac{1}{\mu} + \frac{1-\rho(f(m)/m)}{\rho}\right) = \frac{1}{\lambda}. \tag{18}$$

In the above equation, since the term $f(m)/m$ becomes

$$\begin{aligned}\frac{f(m)}{m} &= \frac{1}{m}\left(m - \frac{1}{m}\sum_{i=1}^{m}\delta_{i-1}\right)\\ &= 1 - \frac{1}{m^2}\sum_{i=1}^{m}\delta_{i-1}\\ &\approx 1\end{aligned}$$

accordingly we have

$$\frac{1}{\mu} + \frac{1-\rho}{\rho} = \frac{f(m)}{\lambda}. \tag{19}$$

By comparing (19) with (17), we can make the important observation that the queueing model shown in Fig. 5 can be changed into that shown in Fig. 6. In Fig. 6, we considered the effect of the restricted rule in terms of the arrival rate, not the service rate.

B. Mean Cell Delay and Average Queue Length

The mean cell delay time is obtained directly from Little's theorem, which is the ratio of the average queue length to the arrival rate. The average queue length can be obtained by differentiating the probability generating function for the steady-state probability distribution. Let the random variable K and $p_k = \Pr[K=k]$ denote the number of cells in a queue just before the arbitration phase and its steady-state probability distribution, respectively. The probability generating function for p_k is defined as

$$G(z) = \sum_{k=0}^{\infty} p_k z^k.$$

TABLE II
$f(m)$ AND $g(m)$ FOR DIFFERENT VALUES OF m WHEN $\rho = 1$. THE RATIO OF $f(m)$ TO $g(m)$ IS ABOUT UNITY FOR ALL m

m	$f(m)$	$g(m)$	$f(m)/g(m)$
1	1.000	1.000	1.000
2	1.618	1.597	1.013
4	2.723	2.614	1.022
8	4.808	4.689	1.025
16	8.871	8.667	1.024
32	16.917	16.121	1.018
64	32.947	32.524	1.013
128	64.967	64.428	1.008
256	128.980	126.833	1.017
512	256.988	255.208	1.007

Then, by differentiating $G(z)$ for z and setting $z=1$, we can obtain the average queue length as in Appendix D:

$$\bar{K} = \frac{\lambda}{f(m)}\frac{(f(m)-\lambda)(2-\lambda)}{\lambda^2 - 2(1+f(m))\lambda + 2f(m)} \tag{20}$$

where $f(m)$ is defined as in (C.3) and the values of $g(m)$ for diverse m are tabulated in Table II. The mean cell delay time is obtained by applying Little's theorem to (20). Then it becomes

$$D = \frac{\bar{K}}{\lambda/f(m)} = \frac{(f(m)-\lambda)(2-\lambda)}{\lambda^2 - 2(1+f(m))\lambda + 2f(m)}. \tag{21}$$

Figs. 7 and 8 show the average queue length and the average cell delay of the MIQ switch using the restricted rule, respectively. They are the functions of m and the average input traffic load as expressed in (20) and (21). Surprisingly, the average queue length is negligibly small even at the higher offered load such as 0.9 when m is larger than 16. That is, the curves have more rectangular elbows as m increases. This phenomenon is accounted for by the fact that the arrival traffic is distributed uniformly to m queues in each input. This implies that the MIQ switch requires much smaller queue size in order to satisfy performance requirements. In the average delay time shown in Fig. 8, curves have more moderate slopes than the average queue length since the queues in an input port are selected with the probability $1/m$.

Additionally, comparing them with those for the MIQ switch using the free rule [41], we can find that both the average queue length and the mean cell delay for the restricted rule are slightly higher than the counterparts for the free rule. The average queue length and the mean cell delay for the free rule can be obtained simply by replacing $f(m)$ in (20) and (21) with m.

C. Cell Loss Probability

The cell loss probability is normally defined as the sum of the cell losses by both a finite buffer space and switching system faults. If we assume that the loss never occurs through a system fault, the cell loss is solely ascribed to the buffer overflow. For the case of infinite buffer size assumed thus far, cell loss probability is upper bounded by the probability of $K > B$, where B

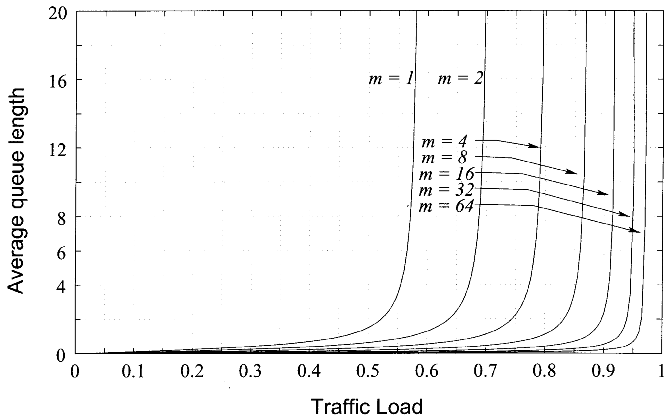

Fig. 7. Average queue length for the MIQ switch with the restricted rule over diverse m.

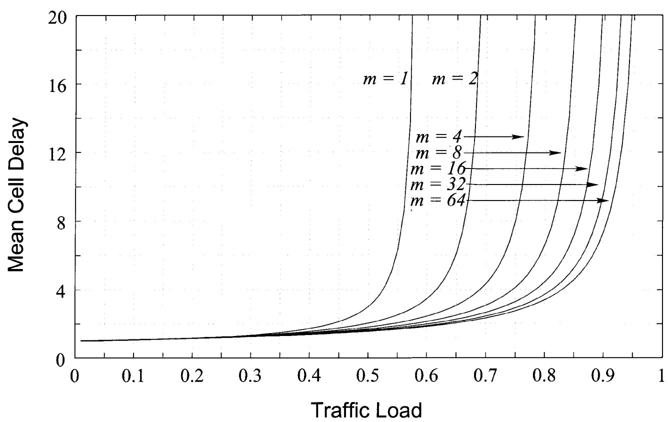

Fig. 8. Mean cell delay for the MIQ switch over diverse m.

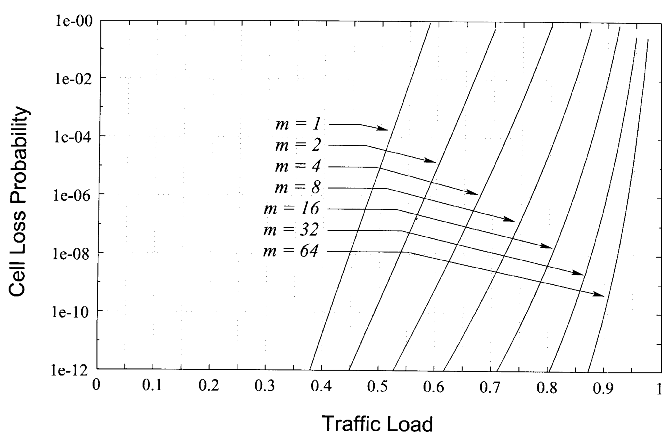

Fig. 9. Cell loss probability for the MIQ switch over diverse m (queue size $B = 16$ cells).

is the finite buffer size. Through the calculation in Appendix E, the upper bound of cell loss probability becomes

$$\Pr[\text{loss}] < \Pr[K > B] = \frac{(2-\lambda)\lambda}{2f(m)(1-\lambda)}\omega_\lambda^B. \qquad (22)$$

Fig. 9 shows this upper bound for different values of m when the buffer size of each queue B is 16 cells. As shown in the figure, the cell loss probability is also negligible as m increases.

V. Conclusion

In this paper, we introduced the concept of the multiple input-queued (MIQ) switch. Differently from the conventional single input-queued (SIQ) switch, which has a single queue in each input, the MIQ switch has $m(1 \leq m \leq N)$ queues in each input. Each of the queues is dedicated to a group of output ports. Since the MIQ switch equips m logically or physically distinct queues in each input, it could serve up to m cells from an input in a time slot. The rule governing this multiple-cell service is referred to as the free rule. However, the multiple-cell service requires the internal speedup in the switch fabric or more sophisticated additional hardware. Therefore, most studies on the MIQ switch assume explicitly or implicitly that at most one cell can be served from an input port. This assumption is referred to as the restricted rule.

Under the free rule, the cell selection for outputs is independent of input ports since multiple cells can be served from an input irrespective of other inputs. Under the restricted rule, however, the cell selection is dependent on other inputs when we assume that the cell selections for all outputs occur sequentially. Therefore, we can evaluate the performance measures in the average form or by using average parameters. In this paper, the saturation throughput, the average queue length and mean cell delay, and the upper bound of the cell loss probability were derived in terms of the number of queues m and the offered load λ. The analysis results show that the saturation throughput approaches 1.0 as m increases, and that the mean cell delay and average queue length and the cell loss probability are negligible even for the higher offered load.

In addition to the enhanced switching performance, the MIQ switch employing the restricted rule operates at the same speed as the external link speed. It implies that the switch is appropriate for high-speed switching systems requiring high performance.

Appendix A
Derivation of $E[N_j]$

The term $E[N_j]$ in (3) is the steady-state expected number of HOL cells for output j in output group 1. Let N_j' be the value of N_j for the next time slot, then N_j' becomes

$$N_j' = N_j - \epsilon(N_j) + A_j \qquad (A.1)$$

in which the random variable A_j is the number of fresh HOL arrivals for output port j, and is independent of N_j for sufficiently large N. Taking expectation of both sides of (A.1) and considering the steady state, we get $E[\epsilon(N_j)] = E[A_j]$ since $E[N_j'] = E[N_j]$ in steady state. Squaring both sides of (A.1), it becomes

$$(N_j')^2 = (N_j)^2 + \epsilon(N_j) + A_j^2 - 2N_j + 2N_jA_j - 2\epsilon(N_j)A_j \qquad (A.2)$$

where the relationships of $\epsilon^2(N_j) = \epsilon(N_j)$ and $N_j\epsilon(N_j) = N_j$ are used. Taking expectation on both sides of (A.2), it becomes

$$0 = E[A_j] + E\left[A_j^2\right] - 2E[N_j] + 2E[N_j]E[A_j] - 2E[A_j]^2$$

since $E[(N'_j)^2] = E[(N_j)^2]$ and $E[\epsilon(N_j)] = E[A_j]$. Arranging it for $E[N_j]$

$$E[N_j] = E[A_j] + \frac{E[A_j(A_j-1)]}{2(1-E[A_j])}. \quad \text{(A.3)}$$

As in [1], A_j becomes Poisson (λ_1) as $N \gg 1$ and, thus, $E[A_j(A_j-1)] \approx \lambda_1^2$. Then (A.3) becomes

$$E[N_j] = \lambda_1 + \frac{\lambda_1^2}{2(1-\lambda_1)}. \quad \text{(A.4)}$$

APPENDIX B
DERIVATION OF $E[N^{(b)}]$

Let R_1 denote the number of unblocked HOL cells of the queues attending the arbitration for output group 1. Then R_1 becomes

$$R_1 = N - N^{(b)} \quad \text{(B.1)}$$

where N denotes the number of queues related to output group 1. Since every cell will eventually become a fresh HOL cell at some time, the flow conservation relationship must hold:

$$E[R_1]\rho = \frac{N}{m}\lambda_1 \quad \text{(B.2)}$$

where ρ, as shown in Fig. 3, is the steady-state probability that a queue has a fresh HOL cell, given that the queue is not blocked during the previous time slot. Taking expectation on both sides of (B.1) and using the flow conservation rule, we have

$$E[N^{(b)}] = \frac{N}{m}\left(m - \frac{\lambda_1}{\rho}\right). \quad \text{(B.3)}$$

APPENDIX C
CLOSED FORM OF THE SATURATION THROUGHPUT

For use in deriving the mean cell delay and the cell loss probability in Section IV, (12) can be expressed in a closed form. Substituting λ_i of (11) into (12), we have

$$\lambda = \frac{1 + \left(m - \frac{1}{m}\sum_{i=1}^{m}\delta_{i-1}\right)\rho - \frac{1}{m}\sum_{i=1}^{m}\mathrm{SQRT}_i}{2-\rho}. \quad \text{(C.1)}$$

When we solved for the throughput of each subswitch, we obtained the throughput expressions by solving the quadratic equations such as (5), (9), and (10). Therefore, we can analogize that λ in (C.1) is one root of the quadratic equation similar to those equations. Then, since one root of the quadratic equation $Ax^2 - 2Bx + C = 0$ $(A \neq 0)$ is obtained by

$$x = \frac{B - \sqrt{B^2 - AC}}{A}$$

we can get a quadratic equation for the average throughput from (C.1) in reverse:

$$(2-\rho)\lambda^2 - 2\left(\left(m - \frac{1}{m}\sum_{i=1}^{m}\delta_{i-1}\right)\rho + 1\right)\lambda + 2g(m)\rho = 0$$

or

$$(2-\rho)\lambda^2 - 2(f(m)\rho + 1)\lambda + 2g(m)\rho = 0 \quad \text{(C.2)}$$

where

$$f(m) = m - \frac{1}{m}\sum_{i=1}^{m}\delta_{i-1}. \quad \text{(C.3)}$$

From the definition of δ_i, we can say that $g(m)$ is also the function of m. However, since $g(m)$ includes nonlinear terms with respect to m, it is not easy to express $g(m)$ in a closed form. Another reason that we do not need to derive $g(m)$ is that $g(m)$ is used neither in the expression of the mean cell delay time nor in the expression of the cell loss probability, as shown in (21) and (22). Therefore, we choose to calculate $f(m)$ and $g(m)$ by using the results of the saturation throughput. Table II shows the values of $f(m)$, $g(m)$, and the ratio of $f(m)/g(m)$ when $\rho = 1$. Notice that the ratio of $f(m)/g(m)$ is around 1.0. Namely, $f(m) - g(m) \approx 0$.

APPENDIX D
DERIVATION OF THE AVERAGE QUEUE LENGTH

The relationship in (19) together with the equation of ρ in terms of λ in (C.2) gives

$$\begin{aligned}\mu &= \frac{2\lambda(\lambda-1)}{\lambda^2 - 2\lambda - 2f(m) + 2g(m)} \\ &= \frac{2(1-\lambda)}{2-\lambda}\end{aligned} \quad \text{(D.1)}$$

since $-2f(m) + 2g(m)$ is nearly zero, as shown in Table II.

Here, let the random variable K be the number of cells in a queue just before the arbitration phase. We shall consider a queue of infinite size for a moment. Then the queue length K' for the next slot is modeled by

$$K' = K + \alpha - \gamma\epsilon(K) \quad \text{(D.2)}$$

in which γ is a Bernoulli random variable with $E[\gamma] = \mu$, the probability that the HOL cell is served, α is also a Bernoulli random variable with $E[\alpha] = \lambda/f(m)$, and the indication function $\epsilon(x)$ is the same as defined in (1).

From (D.2), the probability generating function for the steady-state probability p_k is given by

$$G(z) = E[z^{\alpha}]E[z^{K-\gamma\epsilon(K)}] \quad \text{(D.3)}$$

since $K' = K$ in steady state. The second term on the right-hand side equals

$$\begin{aligned}&E_\gamma\left[\sum_{k=0}^{\infty} p_k z^{k-\gamma\epsilon(k)}\right] \\ &= p_0 + E_\gamma\left[\sum_{k=1}^{\infty} p_k z^{k-\gamma}\right] \\ &= p_0\left[1 - E_\gamma[z^{-\gamma}]\right] + E_\gamma\left[\sum_{k=0}^{\infty} p_k z^{k-\gamma}\right] \\ &= p_0\left[1 - E_\gamma[z^{-\gamma}]\right] + G(z)E_\gamma[z^{-\gamma}]\end{aligned} \quad \text{(D.4)}$$

where $E_\gamma[\cdot]$ is the expectation for the random variable γ. Substituting $E[z^\alpha] = [1 - \lambda/f(m) + (\lambda/f(m))z], E_\gamma[z^{-\gamma}] = [1 - \mu + \mu z^{-1}]$, and (D.4) into (D.3), we have

$$\begin{aligned} G(z) &= \frac{f(m)\mu p_0(z-1)(f(m)-\lambda+\lambda z)}{(\mu-1)\lambda z^2 + ((f(m)-\lambda)\mu - (1-\mu)\lambda)z - (f(m)-\lambda)\mu} \\ &= \frac{f(m)\mu p_0(f(m)-\lambda+\lambda z)}{(f(m)-\lambda)\mu - (1-\mu)\lambda z}. \end{aligned} \tag{D.5}$$

After evaluating p_0 by setting $G(1) = 1$, and substituting p_0 in (D.5), we obtain

$$G(z) = \left(\mu - \frac{\lambda}{f(m)}\right) \frac{(f(m)-\lambda+\lambda z)}{(f(m)-\lambda)\mu - (1-\mu)\lambda z}. \tag{D.6}$$

Differentiating $G(z)$ for z and setting $z = 1$, the expected value of K is given by

$$\bar{K} = \frac{\lambda}{f(m)} \frac{f(m)-\lambda}{f(m)\mu - \lambda}. \tag{D.7}$$

We need to express $\bar{K}$ in terms of λ. Therefore, substituting μ from (D.1) into (D.7) and rearranging the equation in terms of λ, we immediately have

$$\bar{K} = \frac{\lambda}{f(m)} \frac{(f(m)-\lambda)(2-\lambda)}{\lambda^2 - 2(1+f(m))\lambda + 2f(m)}. \tag{D.8}$$

APPENDIX E
UPPER BOUND OF CELL LOSS PROBABILITY

Let us expand (D.6) into a series expansion form as follows:

$$\begin{aligned} G(z) &= \frac{1}{m}(1-\omega_\lambda)\frac{f(m)-\lambda+\lambda z}{1-\omega_\lambda z} \\ &= \frac{1}{m}(f(m)-\lambda+\lambda z)\sum_{k=0}^{\infty}\omega_\lambda^k z^k \end{aligned} \tag{E.1}$$

where $\omega_\lambda = (\lambda(1-\mu)/(f(m)-\lambda)\mu) = \lambda^2/(f(m)-\lambda)(1-\lambda)$. Then, from the definition of $G(z)$, that is

$$G(z) = \sum_{k=0}^{\infty} p_k z^k$$

we have

$$\begin{aligned} p_0 &= (1-\omega_\lambda)\left(1 - \frac{\lambda}{f(m)}\right) \\ p_k &= (1-\omega_\lambda)\left[\left(1 - \frac{\lambda}{f(m)}\right)\omega_\lambda + \frac{\lambda}{f(m)}\right]\omega_\lambda^{k-1} \\ &= \left(\frac{(2-\lambda)\lambda}{2f(m)(1-\lambda)}\right)(1-\omega_\lambda)\omega_\lambda^{k-1}, \qquad k \geq 1 \end{aligned}$$

where $\omega_\lambda = (\lambda(1-\mu)/(f(m)-\lambda)\mu)$ and μ in (D.1) is used. That is, we still use the function $f(m)$ to evaluate the upper bound of the cell loss probability for a specific value of m. We also neglect the term $-2f(m)+2g(m)$ since it has a value near zero. Then, the buffer overflow probability for a finite buffer of size B is defined as the probability that a new cell arrives when the current queue is fully occupied, which is upper bounded by the probability of $K > B$ for the case of infinite buffer size. Therefore

$$\Pr[\text{loss}] < \Pr[K > B] = \left[\left(1 - \frac{\lambda}{f(m)}\right)\omega_\lambda + \frac{\lambda}{f(m)}\right]\omega_\lambda^B.$$

Substituting ω_λ into the above equation, we have

$$\Pr[\text{loss}] < \Pr[K > B] = \frac{(2-\lambda)\lambda}{2f(m)(1-\lambda)}\omega_\lambda^B. \tag{E.2}$$

REFERENCES

[1] J. Y. Hui and E. Arthurs, "A broadband cell switch for integrated transport," *IEEE J. Select. Areas Commun.*, vol. SAC-5, pp. 1262–1273, Oct. 1987.

[2] M. J. Karol, M. G. Hluchyj, and S. P. Morgan, "Input versus output queueing in a space division switch," *IEEE Trans. Commun.*, vol. COM-35, pp. 1347–1356, Dec. 1987.

[3] H. Matsunaga and H. Uematsu, "A 1.5-Gb/s 8 × 8 cross-connect switch using a time reservation algorithm," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1308–1317, Oct. 1991.

[4] M. J. Karol, K. Eng, and H. Obara, "Improving the performance of input-queued ATM packet switches," in *Proc. IEEE INFOCOM*, May 1992, pp. 110–115.

[5] R. Y. Awdeh and H. T. Mouftah, "Survey of ATM switch architectures," *Comput. Networks ISDN Syst.*, vol. 27, pp. 1567–1613, 1995.

[6] M. G. Hluchyj and M. J. Karol, "Queueing in space-division packet switching," in *Proc. IEEE GLOBECOM*, New Orleans, LA, Mar. 1988, pp. 334–343.

[7] ——, "Queueing in high-performance packet switching," *IEEE J. Select. Areas Commun.*, vol. SAC-6, pp. 1587–1597, Dec. 1988.

[8] T. T. Lee, "A modular architecture for very large packet switches," *IEEE Trans. Commun.*, vol. 38, pp. 1097–1106, July 1990.

[9] K. W. Sarkies, "The bypass queue in fast packet switching," *IEEE Trans. Commun.*, vol. 39, pp. 766–774, May 1991.

[10] G. Thomas and J. Man, "Improved windowing rule for input buffered packet switches," *Electron. Lett.*, vol. 29, no. 4, pp. 393–395, Feb. 18, 1993.

[11] S. C. Liew, "Performance of various input-buffered and output-buffered ATM switch design principles under bursty traffic: Simulation study," *IEEE Trans. Commun.*, vol. 42, pp. 1371–1379, Feb./Mar./Apr. 1994.

[12] Y.-S. Su and J.-H. Huang, "Throughput analysis and optimal design of banyan switches with bypass queues," *IEEE Trans. Commun.*, vol. 42, pp. 2781–2784, Oct. 1994.

[13] G. Y. Lee and C. K. Un, "Design and performance analysis of an asynchronous banyan network switch with window policy," *Proc. Inst. Elect. Eng.*, vol. 142, no. 2, pp. 54–60, Apr. 1995.

[14] J.-J. Li, "Improving the input-queueing switch under bursty traffic," *Electron. Lett.*, vol. 31, no. 11, pp. 854–855, May 25, 1995.

[15] W. Warde and P. A. Ivey, "Input queueing multicast ATM packet switch with two priority classes using a priority scheme with a window policy," *Electron. Lett.*, vol. 32, no. 20, pp. 1854–1855, Sept. 26, 1996.

[16] N. J. Sharma, "Comparison of windowing policies for input buffered packet switch," in *Proc. IEEE Int. Performance, Computing and Communication Conf.*, Feb. 1997, pp. 245–251.

[17] D.-Y. Ma, "Performance study of a nonblocking space-division packet switch with window policy," in *Proc. Int. Conf. Information and Communication Security*, vol. 3, Singapore, Sept. 1997, pp. 1410–1412.

[18] I. I. Makhamreh, "Throughput analysis of input-buffered ATM switch," *Proc. Inst. Elect. Eng.*, vol. 145, no. 1, pp. 15–18, Feb. 1998.

[19] X. Ji and D. Wang, "Neural network controller for queueing in packet switch networks," in *Proc. IEEE TENCON*, vol. 3, Beijing, China, Oct. 1993, pp. 207–210.

[20] H. El Gebaly, J. Muzio, and F. El Guibaly, "Input smoothing with buffering: A new technique for queueing in fast packet switching," in *Proc. IEEE Pacific Rim Conf. Communications, Computers, and Signal Processing*, Victoria, BC, Canada, May 1995, pp. 59–62.

[21] A. Huang and S. Knauser, "Starlite: A wideband digital switch," in *Proc. IEEE GLOBECOM*, Dec. 1984, pp. 121–125.

[22] H. Kim, C. Oh, A. Ahmad, and K. Kim, "Performance comparison of high-speed input-buffered ATM switches," in *Proc. IEEE ATM Workshop*, Lisbon, Portugal, May 1997, pp. 505–513.

[23] M. M. Ali and H. T. Nguyen, “A neural network implementation of an input access scheme in a high-speed packet switch,” in *Proc. IEEE GLOBECOM*, vol. 1, Dallas, TX, Nov. 1989, pp. 1192–1196.
[24] M. K. M. Ali and M. Youssefi, “The performance analysis of an input access scheme in a high-speed packet switch,” in *Proc. IEEE INFOCOM*, vol. 1, Bal Harbor, FL, Apr. 1991, pp. 454–461.
[25] E. D. Re and R. Fantacci, “Performance evaluation of input and output queueing techniques in ATM switching systems,” *IEEE Trans. Commun.*, vol. 41, pp. 1565–1575, Oct. 1993.
[26] H. Kim, C. Oh, and K. Kim, “A study on a high-speed ATM switch architecture using random access input-buffers,” in *Proc. Joint Conf. Communications and Information*, vol. 1, Kwangju, Korea, Apr. 1996, pp. 240–244. In Korean.
[27] G. Thomas, “Multi-channel input-queueing for high throughput switches,” *Electron. Lett.*, vol. 33, no. 3, pp. 184–185, Jan. 30, 1997.
[28] G. Nong, J. K. Muppala, and M. Hamdi, “A performance model for ATM switches with multiple input queues,” in *Proc. Int. Conf. Computer Communications and Networks*, Las Vegas, NV, Sept. 1997, pp. 222–227.
[29] G. Thomas and V. Veludandi, “ATM switches with bifurcated input queueing,” in *Proc. Int. Conf. Computer Communications and Networks*, Las Vegas, NV, Sept. 1997, pp. 504–507.
[30] C. Kolias and L. Kleinrock, “Throughput analysis of multiple input-queueing in ATM switches,” in *Broadband Communications*, L. Mason and A. Casaca, Eds, London, U.K.: Chapman & Hal, 1996, pp. 382–393.
[31] G. Thomas, “Bifurcated queueing for throughput enhancement in input-queued switches,” *IEEE Commun. Lett.*, vol. 1, pp. 56–57, Mar. 1997.
[32] K. L. Yeung and S. Hai, “Throughput analysis for input-buffered ATM switches with multiple FIFO queues per input port,” *Electron. Lett.*, vol. 33, no. 19, pp. 1604–1606, Sept. 11, 1997.
[33] G. Nong, J. K. Muppala, and M. Hamdi, “Analysis of nonblocking ATM switches with multiple input queues,” in *Proc. IEEE GLOBECOM*, vol. 1, Phoenix, AZ, Nov. 1997, pp. 531–535.
[34] H. Kim, C. Oh, and K. Kim, “A high-speed ATM switch architecture using random access input buffers and multi-cell-time arbitration,” in *Proc. IEEE GLOBECOM*, vol. 1, Phoenix, AZ, Nov. 1997, pp. 536–540.
[35] H. Kim, C. Oh, and K. Kim, “Throughput analysis of the bifurcated input-queued packet switches with restricted contention,” *Electron. Lett.*, vol. 34, no. 17, pp. 1651–1652, Aug. 20, 1998.
[36] G. Nong, J. K. Muppala, and M. Hamdi, “Analysis of nonblocking ATM switches with multiple input queues,” *IEEE/ACM Trans. Networking*, vol. 7, pp. 60–74, Feb. 1999.
[37] H. Kim, C. Oh, Y. Lee, and K. Kim, “Throughput analysis of the bifurcated input-queued ATM switch,” *IEICE Trans. Commun.*, vol. E82-B, no. 5, pp. 768–772, May 1999.
[38] H. Obara, “Optimum architecture for input queueing ATM switches,” *Electron. Lett.*, pp. 555–557, Mar. 28, 1991.
[39] S.-Q. Li, “Performance of trunk grouping in packet switch design,” in *Proc. IEEE INFOCOM*, vol. 2, Bal Harbor, FL, Apr. 1991, pp. 688–693.
[40] Z. Tao and S. Cheng, “A new way to share buffer-grouped input queueing in ATM switching,” in *Proc. IEEE GLOBECOM*, vol. 1, San Francisco, CA, 1994, pp. 475–479.
[41] H. Kim, C. Oh, K. Kim, and Y. Lee, “Derivation of the mean cell delay and cell loss probability for multiple input-queued ATM switches,” *IEEE Commun. Lett.*, vol. 4, pp. 140–142, Apr. 2000.
[42] H. Kim, H. Yoon, D. Lee, K. Kim, and Y. Lee, “A scheduling algorithm minimizing the FOL blocking in the multiple input-queued switch,” *IEEE Commun. Lett.*, submitted for publication.
[43] T. E. Anderson, S. S. Owicki, J. B. Saxe, and C. P. Thacker, “High-speed switch scheduling for local-area networks,” *ACM Trans. Comput. Syst.*, vol. 11, no. 4, pp. 319–352, Nov. 1993.
[44] N. McKeown, “Scheduling algorithm for input-queued cell switches,” Ph.D. dissertation, Univ. of California, Berkeley, May 1995.
[45] R. O. LaMaire and D. N. Serpanos, “Two-dimensional round-robin schedulers for packet switches with multiple input queues,” *IEEE/ACM Trans. Networking*, vol. 2, pp. 471–482, Oct. 1994.
[46] N. McKeown, V. Anantharam, and J. Walrand, “Achieving 100% throughput in an input-queued switch,” in *Proc. IEEE INFOCOM*, vol. 1, San Francisco, CA, Mar. 1996, pp. 296–302.
[47] A. Mekkittikul and N. McKeown, “A practical scheduling algorithm to achieve 100% throughput in input-queued switches,” in *Proc. IEEE INFOCOM*, vol. 2, San Francisco, CA, 1998, pp. 792–799.
[48] N. McKeown, “The iSLIP scheduling algorithm for input-queued switches,” *IEEE/ACM Trans. Networking*, vol. 7, pp. 188–201, Apr. 1999.
[49] M. Nabeshima and N. Yamanaka, “The i-QOCF (iterative quasi-oldest-cell-first) algorithm for input-queued ATM switches,” in *Proc. IEEE ATM Workshop*, Kochi, Japan, May 1999, pp. 25–30.
[50] S. Y. Liew, S. W. Cheng, and T. T. Lee, “An enhanced iterative scheduling algorithm for ATM input-buffered switch,” in *Proc. IEEE ATM Workshop*, Kochi, Japan, May 1999, pp. 103–108.
[51] R. Schoenen, G. Post, and G. Sander, “Prioritized arbitration for input-queued switches with 100% throughput,” in *Proc. IEEE ATM Workshop*, Kochi, Japan, May 1999, pp. 253–258.
[52] H. Kim, K. Kim, Y. Lee, H. Yoon, and C. Oh, “A simple and efficient cell selection algorithm for the multiple input-queued ATM switch,” in *Proc. IEEE ATM Workshop*, Kochi, Japan, May 1999, pp. 259–264.
[53] H. Takagi, *Queueing Analysis: A Foundation of Performance Evaluation, Volume 3: Discrete-Time Systems*. New York/Amsterdam: Elsevier/North-Holland, 1993, ch. 6.
[54] C. Kolias and L. Kleinrock, “The odd-even input-queueing ATM switch: performance evaluation,” in *Proc. IEEE Integrated Circuits Conf.*, vol. 3, June 1996, pp. 1674–1679.
[55] C. Kolias and L. Kleinrock, “The odd-even ATM switch,” *IEICE Trans. Commun.*, vol. E-81-B, no. 2, pp. 244–250, Feb. 1998.
[56] C. Kolias, “Analysis and performance evaluation of new architecture in high-speed packet switching,” Ph.D. dissertation, Univ. California, Los Angeles, 1999.

Hakyong Kim (A'97) was born in Daejon, Korea, in 1972. He received the B.E. degree in electronics engineering from the Chungnam National University, Daejon, in 1995 and the M.Sc. and Ph.D. degrees in information and communications from Kwang-Ju Institute of Science and Technology (K-JIST), Kwangju, Korea, in 1997 and 2001, respectively.

In 2001, he was with the Ultrafast Fiber-Optic Networks (UFON) Research Center of K-JIST as a Research Professor. He joined Corecess Inc., a leading network equipment company in Seoul, Korea, in December 2001, where he is currently a Senior Engineer. His current research interests are in the area of IP QoS enabling technologies, IP QoS policies, traffic engineering, and network processor architecture. His previous interests include packet switch architecture, queueing strategies and scheduling algorithms and their performance evaluation.

Dr. Kim has been a Technical Programming Committee member of many international conferences, including IEEE ATM'01 and IEEE ICON'01, and has been a reviewer of leading international journals, including IEEE TRANSACTIONS ON COMMUNICATIONS and IEEE COMMUNICATIONS LETTERS.

Kiseon Kim (M'84–SM'98) received the B.Eng. and M.Eng. degrees in electronics engineering from Seoul National University, Seoul, Korea, in 1978 and 1980, respectively, and the Ph.D. degree in electrical engineering–systems from the University of Southern California, Los Angeles, in 1987.

From 1988 to 1991, he was with Schlumberger as a Senior Development Engineer involved in the development of wireless telemetry systems. From 1991 to 1994, he was a Computer Communications Specialist with the Superconducting Super Collider Laboratory, where he built detection and analysis systems for large-scale instrumentation. He joined the Kwang-Ju Institute of Science and Technology, Kwang-Ju, Korea, in 1994, where he is currently a Professor. His current interests include wide-band digital communications system design and analysis and implementation both at the physical layer and at the resource management layer.