Performance Analysis of Asymmetric Packet Switch Modules with Channel Grouping.

Soung Liew and Kevin Lu

Bell Communications Research
445 South Street
Morristown, NJ 07960-1910, U.S.A.
TEL: (201)829-5122

Abstract

This paper analyzes the performance of a class of asymmetric packet-switch modules with channel grouping. The motivation for the study of these switch modules is that they are the key building blocks in large multistage switch architectures. The switch module considered has n inputs and m outputs. A packet destined for a particular output address (out of g) needs to access only one of the r available physical output ports; $m = gr$. Input-buffered, output-buffered, and unbuffered switch modules are studied. Our results show that increasing the number of output ports per output address (r) can significantly improve the performance of buffered as well as unbuffered switch modules. For acceptable performance, the difference in throughput between buffered and unbuffered switch modules is considerable. For buffered switch modules, an interesting observation is that although output-buffered switch modules have significantly better delay performance than input-buffered switch modules when $n = gr$, the performance difference is diminished as we deviate from this switch dimensions.

I. Introduction

This paper considers the performance of the class of asymmetric packet switch modules illustrated in Fig. 1. There are hs input ports consisting of h input groups of s input ports each, and gr output ports of g output groups of r output ports each. The incentive for studying asymmetric switch modules with channel grouping is that one can construct a large switch architecture out of stages of such switch modules. To achieve acceptable performance with the architecture, it is necessary to choose the various parameters of the basic switch modules properly. The objective of this paper is to quantify the performance of these switch modules as a function of the switch dimensions and designs.

Before proceeding further, we give three examples of switch architectures in Fig. 2, 3 and 4 which make use

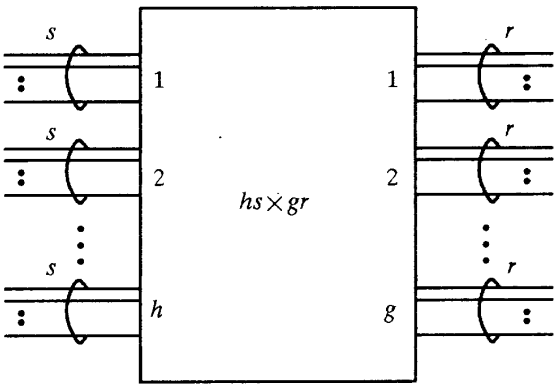


Figure 1: The asymmetric switch module with channel grouping.

of the class of switch modules considered here. Figure 2 is the modular nonblocking switch architecture proposed by Lee [1]. The first stage consists of Batcher-banyan switch modules of dimensions $n \times nk$ (i.e., with respect to the switch module in Fig. 1, $s, r \rightarrow 1$, $h \rightarrow n$ and $g \rightarrow nk$). The second-stage switch modules are statistical multiplexers of dimensions $k \times 1$. Figure 3 is a 3-stage switch architecture proposed in [2]. The dimensions of the first-stage, second-stage, and third-stage switch modules are $n \times m$ ($m > n$), $l \times l'$, and $m' \times n'$ ($m' > n'$), respectively. Here, a channel group of r (r') channels interconnects any first-stage (second-stage) switch module and any second-stage (third-stage) switch module. The structure is such that if r and r' were to be 1, there will be one and only one path between any input and any output. However, in general $r, r' > 1$, and packets have several alternative paths from its input to its destination output. Furthermore, if $m > n$ then the traffic internal to the switch architecture is more "spread-out" than the traffic on the inputs or the outputs, and this

CH2826-5/90/0000/0668/$01.00 © 1990 IEEE

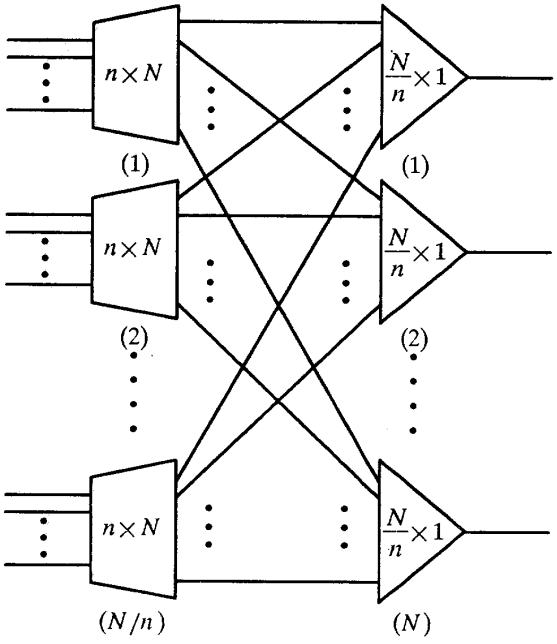


Figure 2: 2-stage nonblocking modular switch architecture.

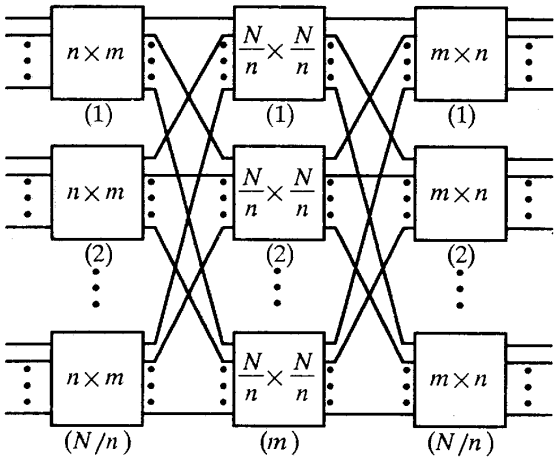
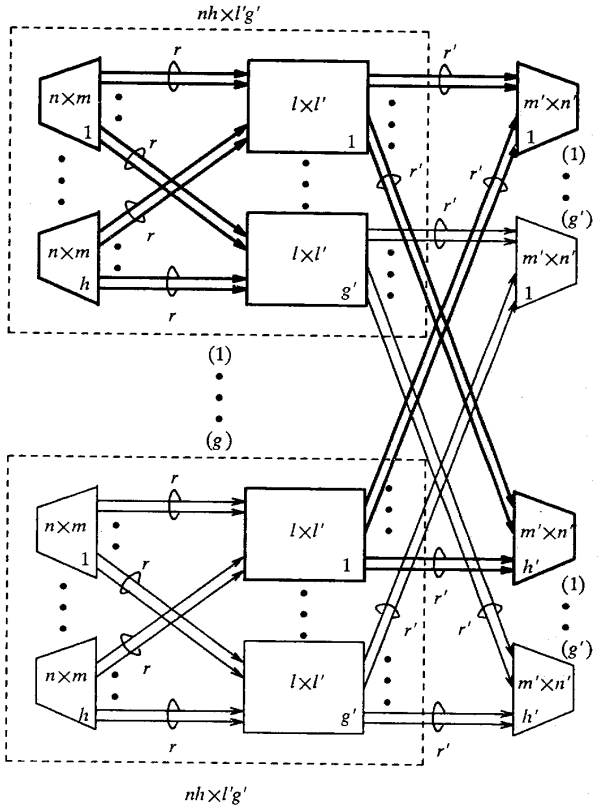


Figure 4: 3-stage Clos switch architecture.



$$N = n\,g\,h = n'\,g'\,h'$$

$$g' = \frac{m}{r} \qquad g = \frac{m'}{r'}$$

$$h = \frac{l}{r} \qquad h' = \frac{l'}{r'}$$

Figure 3: General structure of a 3-stage switch architecture.

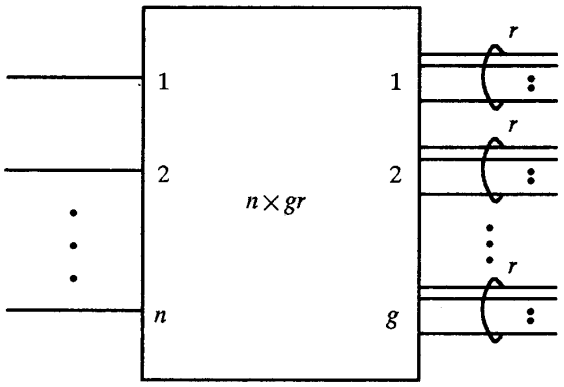


Figure 5: The asymmetric switch module with uncorrelated inputs

leads to better performance of the overall switch. Finally, Fig. 4 is a 3-stage switch architecture [4] that employs asymmetric switch modules at the two outer stages, and symmetric switch modules at the middle stage. There is no channel grouping internally. In all three schemes, asymmetric switch modules at the first stage results in internal line expansion which improves the performance of the overall switch architecture.

Referring to Fig. 1, to the extent that packets at different input ports within the same group are uncorrelated, the switch module reduces to that shown in Fig. 5 in which $hs \rightarrow n$. This paper focuses on the structure shown in Fig. 5, assuming any correlations between packets of different input ports are small and negligible. An output group [3] corresponds to an output address, and a packet can access any of the r output ports of its output address. In any given time slot, at most r packets can be cleared from a particular output group, one packet on each output port. Furthermore, we assume packets are destined for a particular output group (address) rather than a particular output port. That is, it does not matter from which output port a packet exits as long as the output port belongs to the correct output group. For designs of channel-grouping switch modules, please refer to Reference [2]. It turns out that channel-grouping switch modules have smaller complexity (in terms of switch element counts) than ordinary switch modules of the same dimensions.

Three buffering schemes are considered: input queueing, output queueing, and packet dropping [5]. With input queueing, an arriving packet enters a FIFO buffer on its input and waits for its turn to access its destination output. With output queueing, a logical FIFO buffer is allocated to each output group, and arriving packets destined for this output group are immediately placed in it. In any given time slot, at most r packets exit from each output group on the associated r output ports. With packet dropping, no buffers are provided for packet queueing. Any packets not cleared in one time slot are immediately dropped from the system.

For simplicity, we assume homogeneous traffic in our analyses. Packets arrive at input ports with fixed probability in a given time slot, and they have equal probability of being addressed to any output group. Furthermore, there is no correlation between packets arriving in different time slots or at different inputs.

II. Input Queueing

In this section, the maximum throughput of asymmetric input-buffered switch modules with channel grouping is obtained by numerical analysis, and the mean delay by simulations.

It is well known that head-of-line blocking limits the maximum throughput of a symmetric input-buffer to 0.586 [6]. The reader is referred to [5] or [6] for a description of head-of-line blocking. Although the maximum throughput when $n = g$, $r = 1$ and $n \rightarrow \infty$ can be derived by exact analysis, the problem is not amenable to exact analysis when $r > 1$. Nevertheless, a similar approach as in [6] could be taken to a point where the solution could be found by numerical analysis. The same analysis yields the throughput of the switch module as a concentrator ($n > g$) or an expansion network ($n < g$).

To find the maximum throughput, we consider the situation in which the input queues are saturated so that one can always find packets in every queue. In particular, there is always a packet at the head of each queue, waiting to access its destination. Only after this packet is cleared can the next packet move to the head of the queue. Let A_j^i be the number of packets destined for output group i that move into the heads of *free input queues* in the beginning of the j^{th} time slot; the free input queues are queues with packets transmitted in the previous time slot. Let F_{j-1} be the number of free input queues at the end of the $(j-1)^{th}$ time slot, i.e., $F_{j-1} = \sum_{i=1}^{g} A_j^i$. Since each new packet is destined for any output group with equal probability, A_j^i has the binomial probabilities

$$\Pr[A_j^i = k] = \binom{F_{j-1}}{k} \left(\frac{1}{g}\right)^k \left(1 - \frac{1}{g}\right)^{F_{j-1}-k}, \quad k = 0, 1, \ldots, F_{j-1}. \tag{1}$$

At equilibrium, the subscripts can be dropped. As in [6], it can be shown that $\lim_{n,g\to\infty} \Pr[A^i = k] = e^{-\rho_0}\rho_0^k/k!$, where $\rho_0 = \overline{F}/g$. To simplify analysis, we will assume $n, g \to \infty$ while keeping a fixed value of g/n. This approximation is valid when n is large (e.g., $n \geq 16$). The probability generating function (PGF) of A^i is

$$A^i(z) = \sum_{k=0}^{\infty} z^k \Pr[A^i = k] = e^{-\rho_0(1-z)}. \quad (2)$$

Let B_j^i be the number of packets that are destined for output group i at the heads of input queues, but not selected for transmission during the j^{th} time slot. Specifically,

$$B_j^i = \max(0, X_j^i - r) \quad (3)$$

where

$$X_j^i = B_{j-1}^i + A_j^i. \quad (4)$$

Note that X_j^i is the backlog for output group i at the beginning of time slot j, and only when there are more than r packets destined for the same output group will some packets be withheld from transmission. Following a standard approach in queueing analysis [7], we obtain the equilibrium probability generating function

$$B^i(z) = \frac{\sum_{k=0}^{r-1}(z^k - z^r)\Pr[X^i = k]}{A^i(z) - z^r}, \quad (5)$$

where $A^i(z)$ is given in (2). Differenting $B^i(z)$ with respect to z and taking the limit as $z \to 1$ (with L'Hospital's rule applied to remove indeterminacies in the expression for $B^{i'}(1)$), we obtain

$$\overline{B^i} = B^{i'}(1) = \frac{\rho_0^2 - r(r-1)}{2(r-\rho_0)} + \sum_{k=1}^{r-1} \frac{1}{1 - z_k(\rho_0)}, \quad (6)$$

where $\rho_0 = A'(1)$ (i.e., the average number of new packets destined for output group i arriving at the heads of queues per time slot), and 1 and $z_k, k = 1, \ldots, r-1$, are the r zeros of the numerator of $B^i(z)$. It can be shown by using Rouche's Theorem [7] that the denominator of $B^i(z)$ contains exactly r zeros with magnitudes less than or equal to one. Based on the fact that $B^i(z)$ must be analytical for $|z| \leq 1$ (since it is a PGF), these r zeros must also be the r zeros of the numerator. Thus, $z_k, k = 1, \ldots, r-1$, can be found numerically by solving the following $(r-1)$ complex equations

$$A^i(z)^{\frac{1}{r}} - z\left(\cos\frac{2k\pi}{r} + i\sin\frac{2k\pi}{r}\right) = 0,$$

$$k = 1, \ldots, r-1. \quad (7)$$

Note that z_k is a function of ρ_0 because $A^i(z)$ is a function of ρ_0. Now, by symmetry,

$$\overline{B^i} = \frac{1}{g}\sum_{i=1}^{g}\overline{B^i} = \frac{1}{g}(n - \overline{F}) = \frac{n}{g} - \rho_0. \quad (8)$$

Equating (8) with (6), we obtain a equation governing ρ_0 and r. In general, this equation is not in closed form because z_k in turns depends on ρ_0 through (7). But the correct ρ_0 can be found by numerical iteration starting from an initial guess. The maximum throughput per input is related to ρ_0 as follows:

$$\rho^* = \frac{g}{n}\rho_0. \quad (9)$$

For $r = 1$, ρ^* can be expressed in a closed form

$$\rho^* = \left(\frac{g}{n} + 1\right) - \sqrt{\left(\frac{g}{n}\right)^2 + 1}. \quad (10)$$

Table 1 lists the maximum throughput per input for various values of r and g/n. The column in which $g/n = 1$ corresponds to special cases studied by [6] and [8]. For a given r, the maximum throughput increases with g/n because the load on each output group decreases with g/n. For a given g/n, the maximum throughput increases with r because each output group has more output ports for clearing packets. This is analogous to increasing the number of servers in a queueing system. As shown in the table, when g/n is fairly large (say, $g/n > 4$), there is less incentive to use channel grouping to increase the throughput, because the throughput is already close to 1. When g/n is small (say, $g/n < 2$), the use of channel grouping can increase the throughput substantially. For concentrators ($g/n < 1$), increasing the number of output ports per output address from 1 to 2 approximately doubles the maximum throughput.

Table 2 lists the maximum throughput as a function of the line expansion ratio (the ratio of the number of input ports to the number of output ports), $m/n = gr/n$. Notice that for a given line expansion ratio, the maximum throughput increases with r. Channel grouping has a stronger effect on throughput for smaller m/n than for larger m/n. This is because for large m/n, $r = 1$, the line expansion has already alleviated much of the throughput limitation due to head-of-line blocking.

As an example of application of the above results, consider the 2-stage switch architecture in Fig. 2. According to our results, the expanded Batcher-banyan

Table 1: Maximum throughput for an input queue at the first stage with g/n kept constant while $g, n \to \infty$

r	$\frac{g}{n}$										
	$\frac{1}{32}$	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{2}$	1	2	4	8	16	32
1	0.031	0.061	0.117	0.219	0.382	0.586	0.764	0.877	0.938	0.969	0.984
2	0.061	0.121	0.233	0.426	0.686	0.885	0.966	0.991	0.998	0.999	1.000
4	0.123	0.241	0.457	0.768	0.959	0.996	1.000	1.000	1.000	1.000	
8	0.245	0.476	0.831	0.991	1.000	1.000					
16	0.487	0.878	0.999	1.000							
32	0.912	1.000	1.000								

Table 2: Maximum throughput for an input queue at the first stage with m/n (gr/n) kept constant while $m, n \to \infty$.

r	$\frac{m}{n}$					
	1	2	4	8	16	32
1	0.586	0.764	0.877	0.938	0.969	0.984
2	0.686	0.885	0.966	0.991	0.998	0.999
4	0.768	0.959	0.996	1.000	1.000	1.000
8	0.831	0.991	1.000			
16	0.878	0.999				
32	0.912	1.000				
64	0.937					
128	0.955					
256	0.968					
512	0.978					
1024	0.984					

switch modules would have virtually no throughput limitations if $N/n \geq 32$.

Analysis of the mean delay of input-buffered switch modules is difficult, and therefore simulation is used here. Whereas the maximum throughput of input-buffered switch modules is insensitive to the particular contention scheme adopted (as long as no head-of-line packets are withheld for clearance when there are free destination output ports), the mean delay does depend on the contention scheme. One can consider each input queue as a general service FIFO single-server queue, with the service time being the time spent waiting at the head of queue . If the service time and the arrival rate during service are independent, it can be shown that the mean delay is

$$\overline{D} = \overline{S} + \frac{p\overline{S}(\overline{S}-1)}{2(1-p\overline{S})} + \frac{p\mathrm{Var}[S]}{2(1-p\overline{S})}, \qquad (11)$$

where S is the time a packet spends as the head of its queue (i.e., service time) and p is the offered load, or the probability a packet will arrive on a particular input in a given time slot. Thus, $\overline{D}$ depends on the first and second moments of S, which in turn depends on the contention scheme used to arbitrate packets destined for the same output groups. Note that although (11) applies to the random selection policy described in [6], it is is not applicable for the longest queue selection policy in [6], since the service time would then be correlated to the arrival rate during the service.

Arbitration schemes that clear a packet as long as there is a head-of-queue packet would have the same $\overline{S}$. Among these schemes, variations exist as to which packets to clear when there are more than r pack-

ets contending for the sam output group. Among the schemes described by (11), the contention scheme that yields the theoretical minimum mean delay is the one with the minimum Var[S]. To achieve this, our simulation clears the packets that have spent the longest time at the heads of queues. Note that the oldest head-of-queue packets are not necessary the oldest packets within the system, since a packet may have been at the head of queue for a long time, but did not spend much time before reaching the head of queue.

Figure 6 shows the graphs of the mean delay versus the offered load for various values of r and g, fixing n at 32. The number of packets collected for each data point is 5000. Simulation results show that for a given r and g/n, but $n > 32$, the mean delay is closely approximated by the results of $n = 32$. Comparison of the results with [6] shows that, for $g/n = 1$, the mean delay of the oldest head-of-line policy lies between the mean delays of the random and the longest queue selection policies.

III. Output Queueing

For output queueing switch modules, we assume there is a single FIFO queue for an output group. Arriving packets destined for a given output group are immediately placed on the corresponding output queue.

To find the mean delay at the outputs, an approach similar to that used in [6] is taken. Again, let p be the probability that a packet will arrive on a particular input in any given time slot. Let A^i be the number of packet arrivals at a particular output group i during a time slot. Then, A^i has the probability distribution as in (20). The corresponding PGFs are

$$A^i(z) = \left(1 - \frac{p}{g} + z\frac{p}{g}\right)^n. \qquad (12)$$

Let Q_j^i denote the number of packets in a particular output queue at the end of the j^{th} time slot, A_j^i denote the number of packet arrivals during the j^{th} time slot, and X_j^i denote the number of packets in the output buffer in the j^{th} time slot. We find that

$$Q_j^i = \max(0, X_j^i - r), \qquad (13)$$

where

$$X_j^i = Q_{j-1}^i + A_j^i. \qquad (14)$$

Following a standard approach in queueing analysis [7], we obtain the PGF for the steady-state queue size for each individual stage

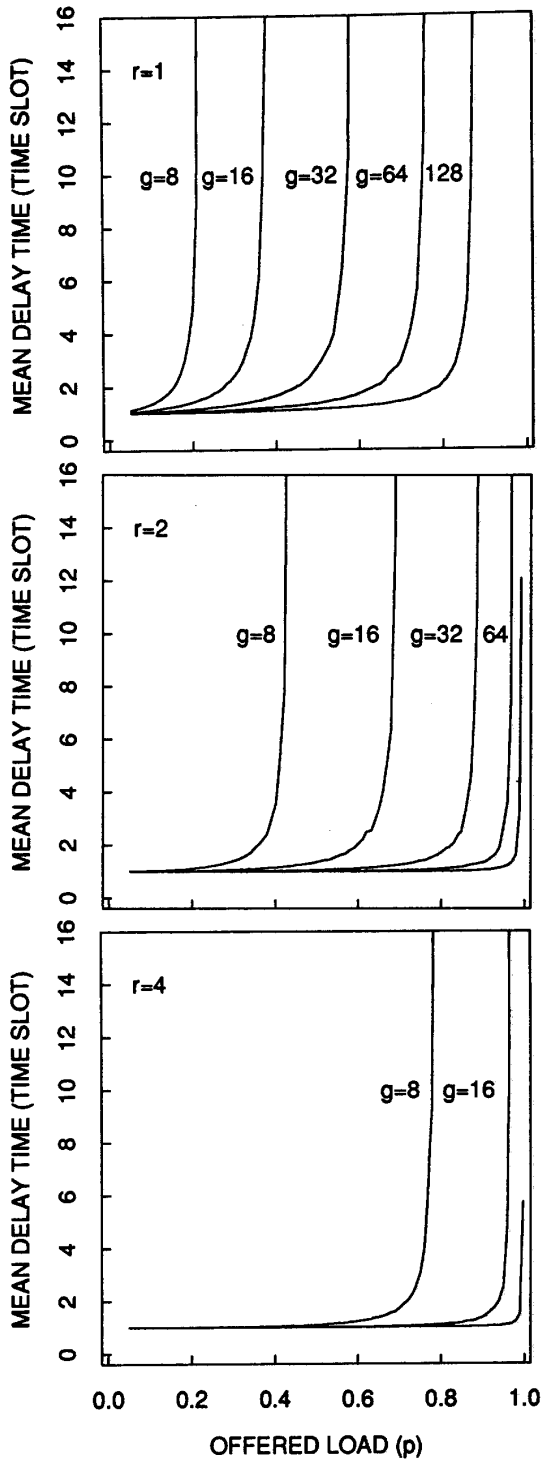


Figure 6: Mean delay vs offered load of input-buffered switch modules

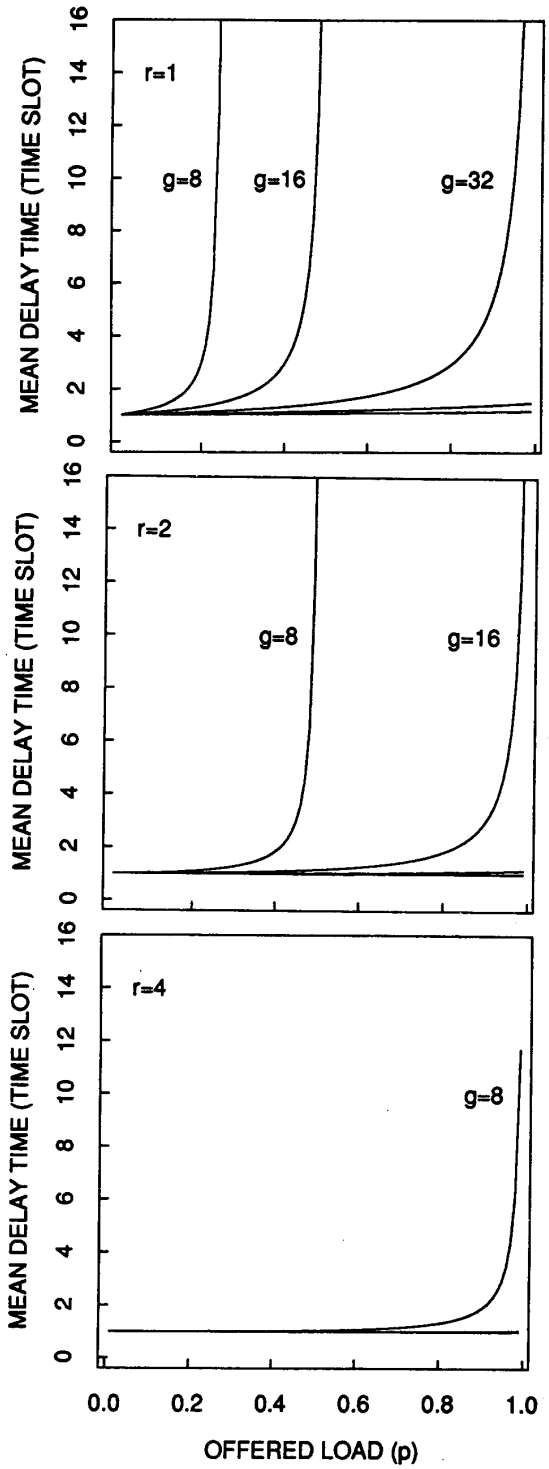


Figure 7: Mean delay vs offered load of output-buffered switch modules

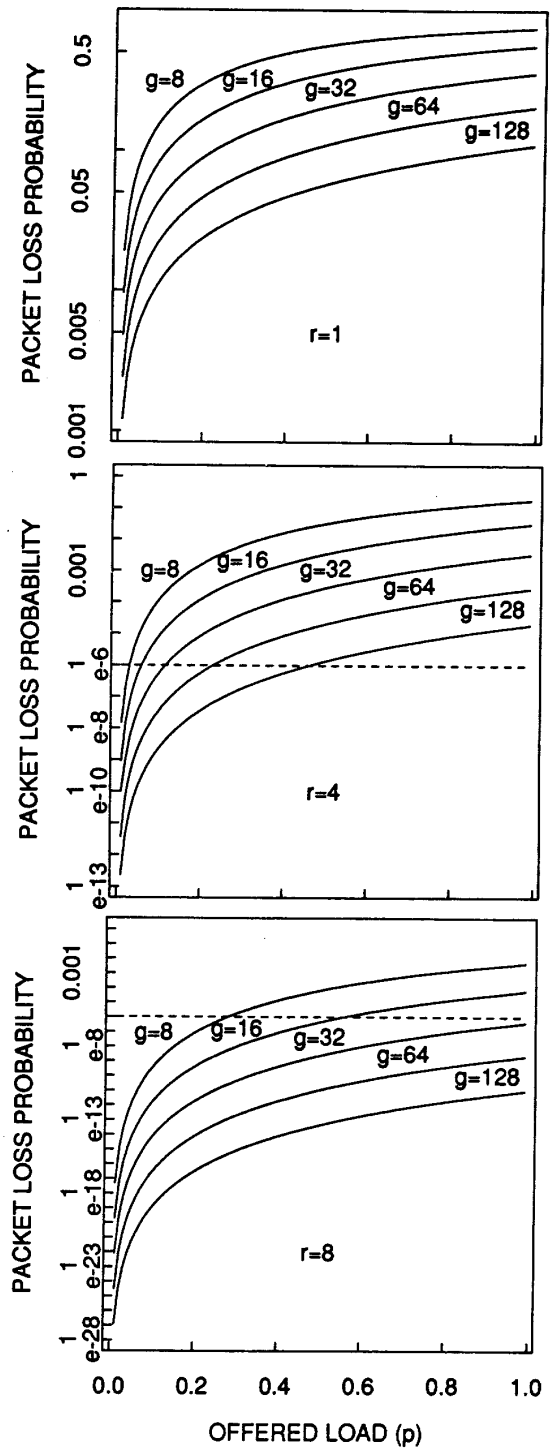


Figure 8: Packet loss probability vs offered load of unbuffered switch modules

$$Q^i(z) = \frac{\sum_{k=0}^{r-1}(z^k - z^r)\Pr[X^i = k]}{\left(1 - \frac{p}{g} + z\frac{p}{g}\right)^n - z^r}. \quad (15)$$

Differenting $Q^i(z)$ with respect to z and taking the limit as $z \to 1$, and using L'Hospital's rule in the expression for $(Q^i)'(1)$, we obtain the mean steady-state queue size

$$\overline{Q^i} = (Q^i)'(1) = \frac{\frac{r(n-1)}{n}\left(p\frac{n}{gr}\right)^2 - r + 1}{2\left(1 - p\frac{n}{gr}\right)} + \sum_{k=1}^{r-1}\frac{1}{1-z_k}, \quad (16)$$

where $z = 1$ and $z_k, k = 1, \ldots, r-1$, are r zeros of the numerator of $Q(z)$. Using the same argument as in Section II, $z_k, k = 1, \ldots, r-1$, are also the zeros of the denominator of $Q^i(z)$ whose magnitudes are smaller than 1. They can be found numerically by solving the following $(r-1)$ complex equations

$$A^i(z)^{\frac{1}{r}} - z\left(\cos\frac{2k\pi}{r} + i\sin\frac{2k\pi}{r}\right) = 0,$$
$$k = 1, \ldots, r-1. \quad (17)$$

where $A^i(z)$ is given in (12).

Using $\overline{Q}$ and Little's Theorem, we obtain the mean waiting time in terms of time slots at each individual stage

$$\overline{W^i} = \frac{\overline{Q^i}}{\left(\frac{np}{g}\right)}. \quad (18)$$

Finally, the mean delay time, $\overline{D^i}$, is simply $\overline{W^i} + 1$.

Now, when n is large, solving for the roots of (17) numerically is difficult because of the polynomial is of very high order. But for a given g/n ratio, we can make an approximation by letting $g, n \to \infty$. Then,

$$A^i(z) = e^{-np(1-z)/g}. \quad (19)$$

Using this $A^i(z)$, the root of (17) can be easily found.

Figure 7 shows the mean delay versus the offered load for various values of r and g, fixing n at 32. The results obtained by assuming a fixed g/n and $n, g \to \infty$ closely follow those presented in the figure.

Comparison of Fig. 6 and Fig. 7 shows that for a fixed r, the improvement of the output queueing performance over the input queueing performance increases as g increases until $gr = n$. Here, we define performance in terms of throughput for a given mean-delay bound (say, mean delay < 2). Beyond $gr = n$, the improvement becomes less distinct. Using the case $r = 1$ as an example, when $g = 8$, there is not much difference between input queueing and output queueing. Intuitively, this is because both input queueing and output queueing are limited by the few number of output ports, and that head-of-line blocking is not the main limiting factor in input queueing switch modules. When $g = 32$, however, the difference is rather distinct, since output queueing is no more limited by line concentration, but input queueing is still limited by head-of-line blocking. Increasing g further does not improve the output queueing performance much since the performance is already near optimum. But the limitation on input queueing is relaxed because of line expansion. In general, for a given r, $gr = n$ is a special case in which there is a substantial difference between the performance of input queueing and output queueing.

IV. Packet Dropping

There is no buffer for packet-dropping switch modules, and whenever there are more than r packets arriving in a time slot, r packets are randomly chosen to clear at the output and the rest are dropped from the system. Using the notation as in the previous section, A^i has the binomial probabilities

$$\Pr[A^i = k] = \binom{n}{k}\left(\frac{p}{g}\right)^k\left(1 - \frac{p}{g}\right)^{n-k},$$
$$k = 0, 1, \ldots, n. \quad (20)$$

The *packet loss probability* or the probability that an arbitrary packet will be dropped from the switch is simply

$$\Pr[\text{packet loss}] = \frac{g}{np}\sum_{k=r+1}^{n}(k-r)\Pr[A^i = k]. \quad (21)$$

Figure 8 shows the packet loss probability versus the offered load for various values of r and g, fixing n at 32. The results obtained by assuming a fixed g/n and and $n > 32$ are closely approximated by those presented in the figure. Using 10^{-6} as the acceptable packet dropping probability, with only 1 port per output address $(r = 1)$, the figure shows that the packet loss probability is unacceptably high over a wide range of switch module parameters: offered load ranging from 0.01 to 1.0, and number of output addresses ranging from 8 to 128. When $r = 4$, there is significant improvement in the packet loss probability, but g would still have to be large for offered loads beyond 0.5. In fact, for $n = g$, r

needs to be at least 8 for the acceptable offered load to go up to 0.5. As in buffered switch modules, for a fixed m/n, performance improves as r increases. Comparing Fig. 8 with Fig. 6 and Fig. 7, it is not surprising that, for a given set of g/n and r values, the acceptable offered load for unbuffered switch modules is very much lowered than for buffered switch modules.

V. Conclusions

This paper has quantified the performance of a class of $n \times gr$ asymmetric packet-switch modules with channel grouping at the outputs. These switch modules constitute the building blocks of many larger switch architectures, and it is important to understand the performance of the switch modules in order to design the larger switch properly. Input-buffered, output-buffered, and unbuffered switch modules have been studied. The performance of the buffered switch modules is defined in terms of the throughput for an acceptable mean delay (say, < 2 time slots), and the the performance of the unbuffered switch modules is defined in terms of the throughput for an acceptable packet loss probability (say, $< 10^{-6}$). In general, for a given set of switch parameters, buffered switch modules have much better performance than unbuffered switch modules. For all cases, however, increasing the number of output ports per output address can significantly improve the performance. For example, for a fixed $g/n < 1$, the throughput of input-buffered switch modules is approximately doubled when we increase the number of output ports per output from 1 to 2. If we fix the line expansion ratio (gr/n) instead, the performance is better for larger r. In other words, decreasing the number of output addresses while fixing the numbers of output and input ports improve the performance. For buffered switch modules, our results show that although output queueing switch modules have significantly better performance than input queueing switch modules when $n = gr$, the advantage is diminished as we increase or decrease g while fixing r. Intuitively, for smaller g, the performance limitation is mainly due to line concentration (i.e., fewer output ports than input ports). But this limitation applies to both input and output queueing switch modules. For larger g, the effect of head-of-line blocking on input queueing switch modules is alleviated because of line expansion, and the performance approaches that of output queueing switch modules. In short, $n = gr$ is a special case in which the difference in performance between input queueing and output queueing is the largest.

Acknowledgements

We thank Tony Lee for generously sharing his knowledge and expertise with us. Howard Lemberg's comments have significantly improved this paper.

References

[1] T. Lee, "A Modular Architecture for Very Large Packet Switches," *Conf. Record, Globecom '89,* vol. 3, pp. 1801–1809.

[2] S. Liew and K. Lu, "A 3-Stage Interconnection Structure for Very Large Packet Switches," *Conf. Record, ICC '90,* pp. 316.7.1–316.7.7..

[3] A. Pattavina, "Multichannel Bandwidth Allocation in a Broadband Packet Switch," *IEEE J. on Selected Areas in Commun.*, vol. 6, no. 9, pp. 1489–99, Dec. 1988.

[4] H. Suzuki *et al.*, "Output-Buffer Switch Architecture for Asynchronous Transfer Mode," *Conf. Record, ICC '89,* vol. 1, pp. 99–103.

[5] M. Hluchyj and M. Karol, "Queueing in High-Performance Packet Switching," *IEEE J. on Selected Areas in Commun.*, vol. 6, no. 9, pp. 1587–97, Dec. 1988.

[6] M. Karol, M. Hluchyj, and S. Morgan, "Input Versus Output Queueing on a Space-Division Packet Switch," *IEEE Trans. on Common.*, vol. COM-35, no. 12, pp. 1347–56, Dec. 1987.

[7] L. Kleinrock, *Queueing Systems, Vol 1: Theory,* Wiley, 1975

[8] Y. Oie *et al.*, "Effect of Speedup in Nonblocking Packet Switch," *Conf. Record, ICC '89,* vol. 1, pp. 410–415.