# Interleaved Multistage Switching Fabrics for Scalable High Performance Routers

Rongsen He and José G. Delgado-Frias
School of Electrical Engineering and Computer Science
Washington State University
Pullman, WA, 99163
{rhe, jdelgado}@eecs.wsu.edu

**ABSTRACT.** As the Internet grows exponentially, scalable high performance routers and switches on backbone are required to provide a large number of ports, higher throughput, lower delay latency and good reliability. At present, most of these routers and switches are implemented on single crossbar as the switched backplane fabric. But the complexity of the single crossbar is increased with $O(N^2)$ in terms of crosspoint number, which is unacceptable for scalability when N becomes large. A delta class self-routing multistage interconnection network with the complexity of $O(N\times\log_2 N)$ has been widely used in the ATM switches. However, the reduction of the crosspoint number results in the serious internal blocking. To solve this problem, quite a few scalable methods have been proposed. One of them, more stages with recirculation architecture is used to reroute the deflected packets, which increase the latency a lot. In this paper, we first bring out the multiple-panel MIN switching fabrics with interleaved recirculation. We also show how to correctly choose the recirculation points to reroute the cells, compared with the wrong connections of former publication. From the simulation under different traffic patterns, this new interleaved architecture, which is insensitive to congestion, could achieve better performance than its counterpart of single panel fabric.

**KEYWORDS.** *Switching fabric, MIN (multistage interconnection network), I-Cubeout network, interleaved recirculation connection.*

## I. INTRODUCTION

Routers construct the skeleton of the Internet. Their kernel, the structure and configuration (scheduler) of the backplane, dominates the routers' performance, scalability, reliability and cost. As higher performance is required with the rapid development of the network economy, router's architecture has also evolved from the shared backplane to switched backplane [1]. At present, most of these routers are based on single stage crossbar, such as the Cisco 12000 series high-end routers. To control the crossbar, complicated schedulers have been developed; for example, PIM [2] was used for DEC's 16-port AN2 switch and iSLIP [3] was used for Cisco 12000 with 16 line cards in a single chassis. So with the complexity of $O(N^2)$ in terms of crosspoint number, these routers just support up to 16×16 interconnection in real applications. It is obvious that a single crossbar has severe limitations to be scalable for next generation network routers, which will incorporate the Internet, telecommunication and TV services. It is required that a large number of line cards should be integrated in a single high performance router.

Multistage interconnection networks (MINs), such as Banyan [4], Omega [5], Baseline and reverse Baseline [6], and indirect binary n-Cube [7], belong to the delta class network which was defined by Patel [8]. They were firstly proposed for large multiprocessor system, which was the hardware foundation of the supercomputers. The purpose is to interconnect processor-to-processors and processor-to-memory modules for fast parallel computation.

The delta class network has two special properties: unique path between each input-output pairs and self-routing in each intermediate stage. Because of the simplicity of its self-routing without a complex scheduler, the delta class network is very attractive for the design of high speed switching fabrics. However, the reduced complexity to $O(N\times\log_2 N)$ also comes with the expense of serious internal blocking, which leads to a poor throughput under some traffic conditions. The basic reason for this is that the delta class is not a permutation network; it could not implement all the permutations of inputs with a single copy ($\log_2 N$ stages) of such a network. In [9], Wu and Feng concluded that $3\times(\log_2 N)-1$ copies through the regular shuffle exchange network are sufficient to realize arbitrary permutation in which N is the network size. A complicated routing algorithm, which realizes the arbitrary permutation, was given at the same time. The algorithm is too complicated to be feasible using a simple and fast hardware implementation. So the prove has just theoretical value. But it shows a direction that more stages will help increase throughput and ease collisions.

In [10] and [11], the authors build the output-queued MIN with b×2b switching elements, so that the number of cells that can be concurrently switched from the inlets to each output queue equals to the number of stages in the interconnection network. In [12], the author improves the architecture in [11] by choosing different recirculation approaches from the last copy of stages. But in order to achieve higher throughput, more stages are required, which increase both the latency and hardware cost.

In this paper, we propose a novel scheme that uses multiple-panel MIN switching fabrics with interleaved recirculation. This novel architecture keeps the SE hardware complexity acceptably low and achieves better performance under various traffic patterns. The rest of the paper is organized as follows. Section 2 describes the details of this new architecture of switching fabrics. To evaluate this architecture, extensive simulation results and analysis are presented in Section 3. Finally, Section 4 concludes the paper.

1-4244-0357-X/06/$20.00 ©2006 IEEE
*This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2006 proceedings.*

## II. INTERLEAVED MULTISTAGE SWITCHING FABRICS

Because our switching fabrics architecture consists of small b×2b crossbar switching elements (SEs), it is crucial to make an assumption throughout the paper that these crossbars are fixed length cell based, though the cell length here is not necessarily as that of ATM cells (53 bytes). Other researchers have also used "packet" as term [12]. But in OSI or TCP/IP model, "packet" always means variable length PDU (protocol data unit). So we still borrow the name "cell" from ATM just as in [1] and [13]. From the perspective of hardware design and fair scheduler in each local SE, processing fixed length cells is much simpler and efficient than handling the variable length packets [1]. Each stage of the switching fabric can be synchronized with the same clock signal and move the cell to next stage or local outlets at the same time.

Variable length packets must be segmented into the fix-sized cells before being transferred across the switching fabric. At the output, the cells are reassembled into previous packets before being sent to outgoing LC. In [12], Tzeng gives a mechanism to keep track the cells in transmission for resequencing at their destinations. So we will not cover the packet segmentation and reassemble in this paper.

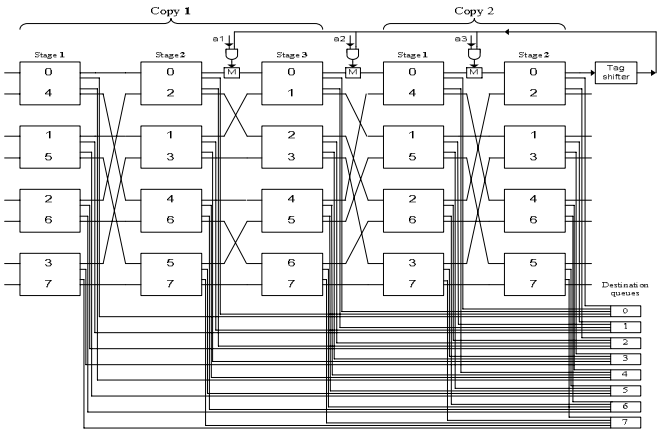


Figure 1. $ICO_8$ with recirculation.

### A. Single panel multistage switching fabric

Based on research published in [10] and [11], Tzeng [12] proposed a new multistage switching fabric for scalable routers as shown in Figure 1, which is based on I-Cubeout (ICO). Each b×2b SE has b remote outlets to connect to next stage and b local outlets to terminate the cells from the switching fabric to the destination queues. Adjacent stages are interconnected according to the indirect n-cube connecting patterns [7]. Let $L$ be the index of line and be expressed in binary notation as follows:

$$L = 2^{n-1} l_1 + 2^{n-2} l_2 + \cdots + 2 l_{n-1} + l_n \qquad (1)$$

(Where $n=log_2N$, $N$ is the network size)

So the indices of the lines incident on the SE on either side differ only in $l_k$. Specifically, the two indices of any SE in the same stage differ by a constant; those in stage 1 differ by N/2, those in stage 2 differ by N/4 and so on. A full copy of the indirect n-cube network consists of all $\log_2 N$ stages. However, the ICO may contain any number of stages, but at least one full copy of stages. The stage i after the first full copy just repeats the (i mod ($\log_2 N$) +1) in a similar fashion as in Figure 1.

The self-routing method of the ICO is slightly different from that used in normal delta class network, as shown in Figure 2 without the recirculation path and output logic for clarity. At the primary input (input of first stage), the routing tag of each cell is generated by bit-wise XOR of the local primary input address and its destination address. If a tag bit corresponding to stage i is "1", the cell needs to take the "cross" state of SE at stage i of any copy. After the non-zero tag bit is corrected this way, it will be reset to "0". If a tag bit corresponding to stage i is "0", the cell just passes straight through the SE of the corresponding stage. When all the tag bits become "0", that means the cell has reached to its destined row and may take the associated local outlet at the SE to its destination queue, through which the output LC is connected.

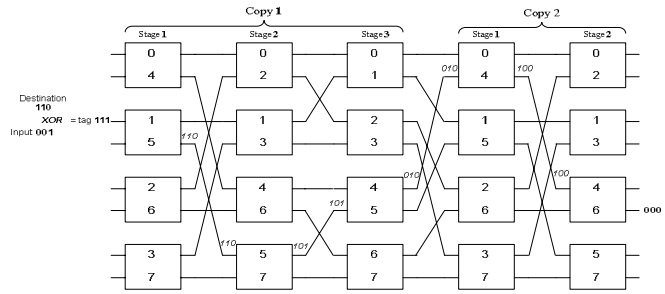


Figure 2. Routing of the switching fabric.

For simplicity of hardware design, the routing tag of the cell will be cyclic rotated leftward by $\log_2 b$ bits after the cell advances to next stage. So, only the leftmost tag bits are examined at each SE. This will unify the design of SE without correlating it with the stage number in which the SE is located. The distance of the cell is defined as the rightmost nonzero bit position $q$ of its tag, which means that the cell still needs to travel at least $q$ stages before getting to its destination queue.

The local scheduler of each SE follows the shortest path algorithm. Two cells, which have different tag bit for the same SE, will conflict with each other (cross and straight through requests). The local SE scheduler should give the priority to the cell with smaller distance and deflect another one, so as to keep cells in the switching fabric as few as possible and improve the system performance. If both have identical distance, a random one is chosen for priority.

In the example shown in Figure 2, a cell with destination 110 comes into input 001. Then a tag 111 will be generated by the XOR operation. Next at stage 1 of copy 1, the cell cross the SE to clear the leftmost bit accordingly and cyclic shift one tag bit left. The new tag is 110. At stage 2 of copy 1, the cell is deflected and cyclic shift one tag bit left, so the corresponding bit still keeps "1". Finally, the cell arrives at destination 110 at stage 2 of copy 2 with all zero tag.

### B. Recirculation connection

In order to use the switching fabric more efficiently and improve the system performance with limited hardware resources, we could reenter the cells, which failed to get to their destination queue after the primary output (the output of

1-4244-0357-X/06/$20.00 ©2006 IEEE
*This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2006 proceedings.*

the last stage), into the last copy of switching fabric again by recirculation (to avoid intensive collisions at previous copies). In [12], Tzeng proposes three approaches for choosing the reentry point shown in Figure 1: static connection, FA (first available point) and FO (first "1" bit in routing tag). When there is no cell arriving from the prior stage to the reentry point concurrently, the recirculated cells can be fed into the switching fabric through the multiplexers.

In [12] and the simulator used for it, the recirculation just connects back to the same physical row (physical row means the identical row in the real topology) as shown in Figure 3. We believe this is not correct; to show this we have a counter example. A cell with destination "110" comes into input "001", and then routing tag "111" is generated. We assume that the cell is deflection-routed in stage 1 and 2 just as label on the path. In stage 3, the correct path is chosen, which cleared the leftmost tag bit and cyclic rotated it to rightmost position. In stage 4, the cell is deflection-routed again and chooses the correct path in stage 5. At the primary output 2, the cell will be recirculated through multiplexer M2 by FO or by FA approach (if M1 is not available). Finally, the cell is routed correctly in stage 4 and all tag bits are cleared to zero, as shown with the tags in circle. That means the cell has reached its destination queue and should be extracted from the switching fabric. But the local real address is "000" and "100", the cell will never have a chance to reach its correct destination with a all zero tags! This routing failure is because of the recirculation to the same physical row, which changes multiple values of $l_i$ for index in equation (1).

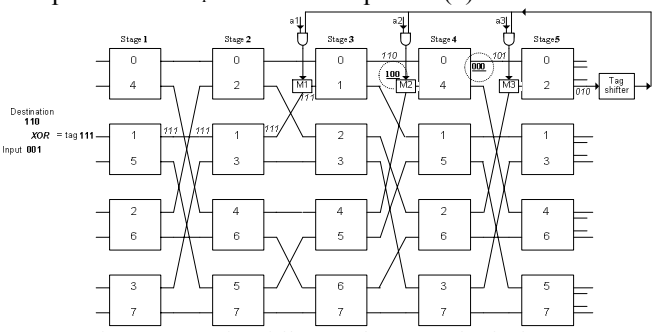


Figure 3. Routing failure to the same physical row.

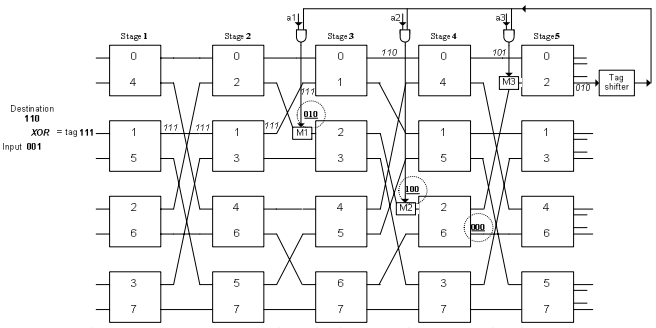


Figure 4. Correct recirculation to the same logic row.

So the correct recirculation should always connect to the same logic row (logical row means the same row which has identical index of lines) as shown in Figure 4. The cell is fed to either M1 or M2. The cell will be terminated at the correct destination queue as shown by the tags in circle.

### *C. Interleaved multistage switching fabrics*

Parallel Banyan network or replicated Delta network [14] was brought out many years ago. But after splitting the flows at the input multiplexers, the flows will be separated independently through each panel of switching fabric. So sometimes the expensive hardware resource is not fully utilized in case of unbalanced traffic patterns. Combined with single panel of multistage switching fabric above, we propose a new architecture of interleaved multistage switching fabrics as shown in Figure 5.

We glue together multiple panels of ICO network shown in Figure 2, both at the primary inputs and outputs. At the N inputs, N demultiplexers distribute the input traffic into each panel synchronized with a clock. At clock cycle t, all the input cells at that time will enter the panel (t mod P), for total P panels from 0 to P-1. At the primary outputs of each panel, we use the recirculation connections as shown in Figure 4 to reroute the deflection-routed cells into the switching fabric of next panel in modular. So the recirculation flows of panel i will go to panel ((i+1) mod P). However, the recirculation entry points still follow the logic rows as shown in Figure 4, even though to different panels. A concentrator is located before each destination queue for terminating the cells from the local outlets of SEs. With the speedup $\xi$, each concentrator could choose up to $\xi$ cells in one clock cycle, from the active rightmost to leftmost stages independent of panels. We give higher priority to rightmost outlets in order to avoid starvation for recirculated cells in case of bursty flows.

After the multiple switching fabrics are interleaved by the recirculation, the scheme provides another opportunity to balance the cell traffic; this in turn effectively eases the hot flows after collision. With the comparison to the single panel of switching fabric in Figure 1, our interleaved architecture should achieve better performance because the flows are balanced and switched in parallel. In next section, we evaluate this scheme's performance through simulations under different traffic patterns.

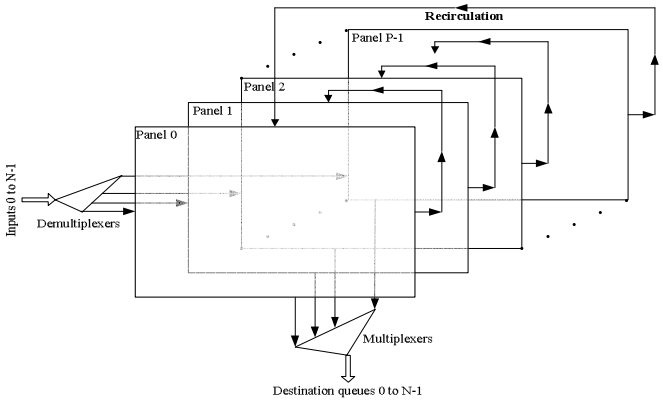


Figure 5. Interleaved multistage switching fabrics.

## III. PERFORMANCE EVALUATION AND RESULTS

We use simulation to evaluate the performance of our interleaved switching fabrics. Professor Tzeng made available to us the simulator used in [12]; the recirculation bug shown in Figure 3 was corrected. The simulator was modified to fit the

requiremenst of our architecture of interleaved switching fabrics. To compare with results of [12], we also choose 256 inputs/outputs with 4×8 SEs.

### *A. Simulation model*

The cell flows are fed into each panel through the 256 demultiplexers at the inputs. So at clock t, the input flows coming at that clock will go to panel (t mod P). The offered load p is defined as the probability that a cell is generated at each input during one cycle. In the simulations, two panels are used for simplicity and practical purposes, so P=2. If P>2, the marginal gain of performance over hardware cost degrades with saturation, so we do not show the simulation results here.

Tzeng [12] has shown that buffered fabrics will get better performance, so we also use the same model for comparison. Each SE output queue (either local or remote one) is equipped with 12-cell buffers. At the outputs, each destination queue could runs with a speedup $\xi$. Up to the capacity, $\xi$ cells can go through the outputs to outside LC. As the simulation in [12], we also choose $\xi$= 2.

The cells take one system cycle to move from one stage to the next stage, as same as that the deflection-routed cell use to reenter the switching fabrics through the recirculation. In [12], the author use $ICO^{FA}$ (first available point) and $ICO^{FO}$ (first "1" bit in routing tag) for better performance rather than $ICO^{S}$ (static recirculation connection). Though $ICO^{FO}$ is a little better than $ICO^{FA}$ from their simulation, it needs to detect the first "1" bit from the left end in each cell's routing tag, which requires much more complicated hardware and delays the whole system. By contrast, $ICO^{FA}$ gets the information of availability directly from prior SE's outlet latch indicator, which eases the hardware design and improves system speed. So we choose the $ICO^{FA}$ as the recirculation connection in our simulation. For all results, 200,000 system clocks are simulated, which are long enough to get steady state results.

### *B. Performance under uniform traffic patterns*

Under the uniform traffic, the cells at the inputs choose each destination output with equal probability. The mean latency versus offered load under single panel P=1 and interleaved double panels P=2 is shown in Figure 6. For P=1, we test it with 4, 6, 8 and 12 stages. For P=2, stages 4, 6 and 8 per panel are simulated. In the figures and below, we use the notation SX/PY where X specifies the number of stages and Y the number of panels. S8/P1 has identical number of SEs as S4/P2, just as S12/P1 with S6/P2. In terms of stages, S4/P1 and S4/P2 (as S6/P1 and S6/P2, S8/P1 and S8/P2) have the same length.

From Figure 6, it can be observed that the interleaved fabrics significantly reduce mean latency (4.7 cycles of P=2 over about 16 cycles of P=1 at load p=1.0). Because of its parallel switching fabrics, the proposed scheme shows considerably less latency degradation with the increase of uniform traffic. Figure 7 shows the drop rate versus offered load. With the same length of stages, S4/P1 and S6/P1 have much larger drop rate than S4/P2 and S6/P2 respectively, in these two comparisons the hardware is doubled. But when comparing S6/P2 and S12/P1, we find that slight increase of drop rate (0.013% of S6/P2 over 0 of S12/P1). These two configurations, S6/P2 and S12/P1, have a significant difference (about 16/4.7=3.4 times) in their mean latency; however, S6/P2 is preferred for real time connectionless applications. Even for connection-oriented applications, TCP's ARQ (Automatic Repeat Request) scheme compensates for the negligible drop rate.

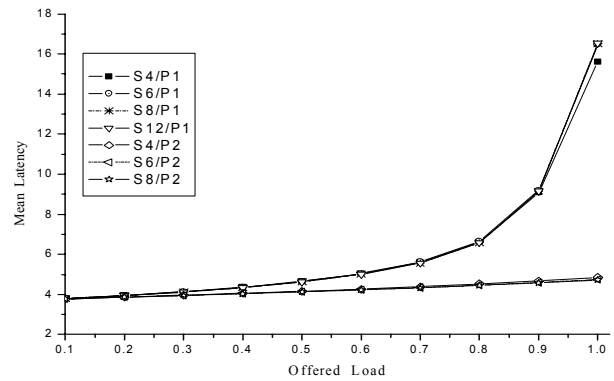


Figure 6. Mean latency vs offered load under uniform traffic.

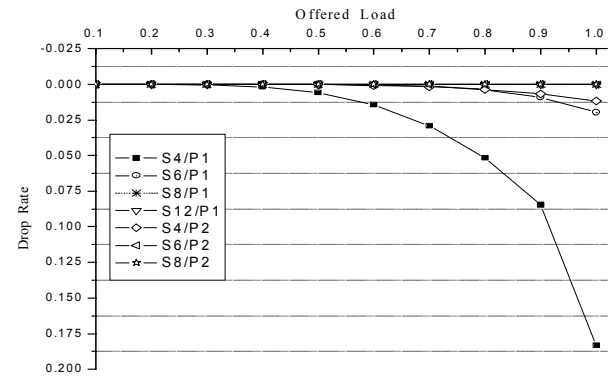


Figure 7. Drop rate vs offered load under uniform traffic.

### *C. Performance under hot-spot traffic patterns*

Traffic over the switching fabrics is usually nonuniform. There are always some hot spots on the network, such as file servers, popular web sites, and uplink to backbone network. In [12], Tzeng uses a single hot spot model to measure the fabric performance under nonuniform patterns. The hot spot model will have more pronounced traffic congestion. For comparison purpose, we also use the single hot-spot model for simulation as shown in Figure 8 and 9.

The hot spot, which is chosen at output port 0, collectively receive $\eta$=10 percent hot traffic in addition to its fair share of 90 percent regular traffic left. The 90 percent regular traffic is evenly distributed over all 256 output ports. From Figure 8, the interleaved fabrics still show great advantage with much lower mean latency against load increase (less than 8.2 cycles of P=2 over more than 18 cycles of P=1 at load p=1.0). In single panel group of P=1, the latency increases with more stages in each panel, because the rightmost stages have higher priority to the destination queues than the leftmost ones in Figure 5. In double panel group of P=2, the results become complicated: S4/P2 has larger mean latency at p=1.0. Due to

1-4244-0357-X/06/$20.00 ©2006 IEEE
*This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2006 proceedings.*

S4/P2's single copy of fabric in length, the intense congestions will deflect more cells after collision. However, deflection is expensive for the hardware resource. You need to correct the deflected tag bit back again in the following stages, which increases the overall mean latency.

In Figure 9, S4/P1 still has the highest drop rate, because there is single copy of I-Cube network and the recirculated cells have many collisions with the cells just from inputs. Because S4/P2 balances the recirculated cells between each panels and decouples them with the input flows, it decrease the drop rate from 33% of P=1 to 12.2% at load p=1.0. The same phenomenon happens between S6/P1 with S6/P2. With the same hardware resource of SEs, S6/P2 just has a little higher drop rate 10.2% over 9.6% of S12/P1. Considering S6/P2's mean latency is just 5.9 clocks over S12/P1's mean latency of 20.8 clocks at p=1.0, our simulations have shown that the interleaved fabrics scheme outperforms its single panel counterpart, just as it does under uniform traffic patterns. Due to $\xi$= 2 and minus the uniform traffic, the theoretical drop rate at full load should be: 10%(hot traffic)-1.1×0.4%(1hot ports/256ports)≈9.56%. Thus the simulation results around 9.6% of S12/P1 and S8/P2 match with this theoretical value.

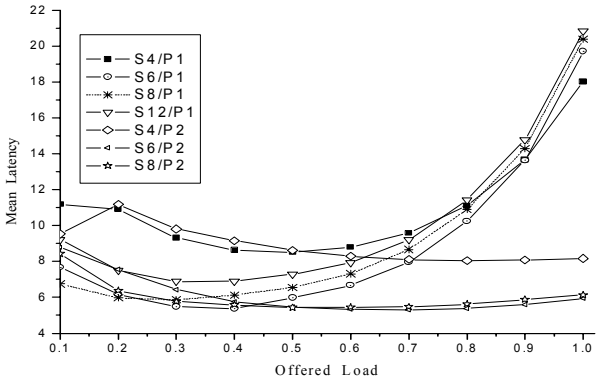


Figure 8. Mean latency vs offered load under hot-spot traffic.

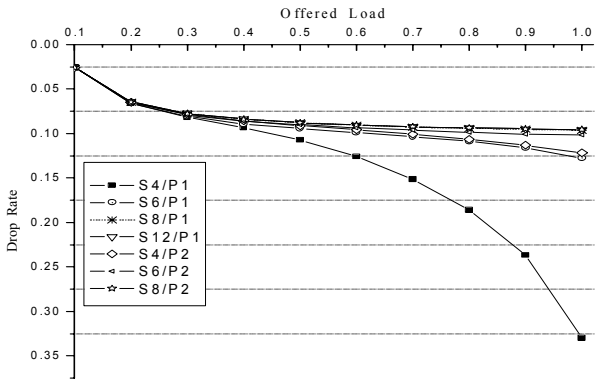


Figure 9. Drop rate vs offered load under hot-spot traffic.

## IV. Concluding Remarks

In this paper, we presented a novel architecture of interleaved switching fabrics for scalable high performance routers. Simulations under different traffic patterns have shown that the interleaved switching fabrics are insensitive on mean latency against load congestion because of its parallel switching. The mean latency deteriorates considerably for the single panel fabric after increase of the load. So this property against congestion is highly preferred for backbone routers with a number of real time applications (such as VOIP, network conference and games).

With the same length of stages, S4/P2 (S6/P2 and S8/P2) always outperform (latency, throughput) their pairs S4/P1 (S6/P1 and S8/P2) under different traffic patterns. Although the number of SEs increases linearly, the proposed scheme is very promising and scales well with present technology as compared to the exponential increase of single crossbar. We have shown that even with the same number of SEs under different organizations, S6/P2 exhibits better performance than S12/P1 (around 3 times in average latency) except its negligibly higher drop rate. With a little increase of hardware resources, S8/P2 outperforms S12/P1 under any traffic patterns. Moreover the interleaved multistage switching fabrics reveal to be highly fault tolerant against internal link or SE failures.

## References


[1] N. McKeown, "Fast Switched backplane for a Gigabit switched router," White paper, http://www.cisco.com

[2] T. Anderson, S. Owicki, J. Saxe, and C. Thacker, "High speed switch scheduling for local area networks," *ACM Trans. Computer Systems,* Vol. 11, No. 4, pp. 319-352, Nov. 1993.

[3] N. McKeown, "The iSLIP scheduling algorithm for input-queued switches," *IEEE/ACM Transactions on Networking*, Vol. 7, No. 2, pp. 188-200, April 1999.

[4] L. R. Goke and G. J. Lipovski, "Banyan networks for partitioning processor systems," *Proc. 1st Annual Symp Computer Architecture,* pp. 21-28, Dec. 1973.

[5] D. Lawrie, "Access and alignment of data in an array processor," *IEEE Transactions on Computers*, Vol. 24, No. 12, pp. 1145-1155, Dec. 1975.

[6] C.-L. Wu and T.-Y. Feng, "On a class of multistage interconnection networks," *IEEE Transactions on Computers*, Vol. 29, No. 8, pp.694-702, Aug. 1980.

[7] M. C. Pease, "The indirect binary n-Cube microprocessor array," *IEEE Transactions on Computers*, Vol. 26, No. 5, pp. 458-473, May 1977.

[8] J. H. Patel, "Performance of processor-memory interconnections for multiprocessors," *IEEE Transactions on Computers*, Vol. 30, No. 10, pp. 771-780, Oct. 1981.

[9] C.-L. Wu and T.-Y. Feng, "The universality of the shuffle-exchange network," *IEEE Transactions on Computers*, Vol. 30, No. 5, pp. 324-332, May 1981.

[10] S. Bassi, M. Decina, P. Giacomazzi, and A. Pattavina, "Multistage shuffle networks with shortest path and deflection routing for high performance ATM switching: The open-loop Shuffleout," *IEEE Transactions on Communication,* Vol. 42, No. 10, pp. 2881-2889, Oct. 1994.

[11] M. Decina, P. Giacomazzi, and A. Pattavina,, "Multistage shuffle networks with shortest path and deflection routing for high performance ATM switching: The closed-loop Shuffleout," *IEEE Transactions on Communication,* Vol. 42, No. 11, pp. 3034-3044, Nov. 1994.

[12] N.-F. Tzeng, "Multistage-based switching fabrics for scalable routers," *IEEE Transactions on Parallel and Distributed Systems,* Vol. 15, No. 4, pp. 304-318, April 2004.

[13] N. Ni and L. N. Bhuyan, "Fair scheduling in Internet routers," *IEEE Transactions on Computers*, Vol. 51, No. 6, pp. 686-701, June 2002.

[14] C. P. Kruskal and M. Snir, "The performance of multistage interconnection networks for multiprocessors," *IEEE Transactions on Computers*, Vol. 32, No. 12, pp. 1091-1098, Dec. 1983.