Output-Based Shared-Memory Crosspoint-Buffered Packet Switch for Multicast Services

Ziqian Dong and Roberto Rojas-Cessa

***Abstract*— The incorporation of broadcast and multimedia-on-demand services are expected to increase multicast traffic in packet networks, and therefore in switches and routers. Combined input-crosspoint buffered (CICB) switches can provide high performance under uniform multicast traffic, however, at the expense of N^2 crosspoint buffers. In this letter, we propose an output-based shared-memory crosspoint-buffered (O-SMCB) packet switch where the crosspoint buffers are shared by two outputs and use no speedup. The proposed switch provides high performance under admissible uniform and nonuniform multicast traffic models while using 50% of the memory used in CICB switches. Furthermore, the O-SMCB switch provides higher throughput than an SMCB switch with buffers shared by inputs, or I-SMCB.**

***Index Terms*— Multicast, buffered crosspoint, buffered crossbar, shared memory, packet switch.**

I. Introduction

The migration of broadcasting and multicasting services, such as cable TV and multimedia-on-demand to packet-oriented networks is expected to take place in the near future. These highly popular applications have the potential of loading up the next generation Internet. To keep up with the bandwidth demand of such applications, the next generation of packet switches and routers need to provide efficient multicast switching and packet replication.

A lot of research has focused on unicast traffic, where each packet has a single destination. It has been shown that unicast switches achieve 100% throughput under admissible conditions, $\sum_i \lambda_{i,j} < 1$ and $\sum_j \lambda_{i,j} < 1$, where i is the index of inputs ($0 \leq i \leq N-1$), j is the index of outputs ($0 \leq i \leq N-1$) for an $N \times N$ port switch, and $\lambda_{i,j}$ is the data rate from input i to output j, in a plethora of switch architectures and switch configuration schemes.

Although it is difficult to describe actual multicast traffic models, switches also need to provide 100% throughput under admissible multicast traffic. In multicast switches, the admissibility conditions are similar to those for unicast traffic, however, with the consideration of the fanout of multicast packets. The fanout of a multicast packet is the number of different destinations that expect copies of the packet. This implies that the average fanout of multicast traffic increases the average output load of a switch. Therefore, the average output load in a multicast switch is proportional to the product of the average input load and the average fanout for a given multicast traffic model.

Here, we consider having incoming variable-size packets being segmented into fixed-length packets, also called cells, at the ingress side of a switch and being re-assembled at the egress side, before the packets leave the switch. Therefore, the time to transmit a cell from an input to an output takes a fixed amount of time, or time slot.

Herein, we consider that cell replication is performed at the switch fabric by exploiting its space capabilities [1]. We focus on crossbar-based switches. Therefore, multicast cells can be stored in a single queue at the input.

Multicast switching has been largely considered for input buffered (IB) switches. In these switches, matching has to be performed between inputs and outputs to define the configuration on a time-slot basis. This matching process can be complex when considering multicast traffic. Combined input crosspoint-buffered (CICB) packet switches have shown higher performance than IB switches at the expense of having line-speed running crosspoint buffers under unicast traffic. In these switches, an input has N virtual output queues to avoid head-of-line blocking [2]. The crosspoint buffers in CICB switches can provide call splitting (or fanout splitting) intrinsically. CICB switches do not use matching as IB switches do [3]-[7]. In CICB switches, one input can send up to one (multicast) cell to the crossbar, and two or more cells destined to a single output port can be forwarded from multiple inputs to the crossbar at the same time slot [8], [9]. Therefore, CICB switches have natural properties favorable for multicast switching. However, CICB switches have dedicated crosspoint buffers for each input-output pair, for a total of N^2 crosspoint buffers. Since memory used in the crosspoint buffers has to be fast, it is desirable to minimize the amount of it as fast memory is expensive.

In response to this need, we propose an output-based shared-memory crosspoint-buffered (O-SMCB) packet switch. This switch requires less memory than a CICB switch to achieve comparable performance under multicast traffic and no speedup. Furthermore, the O-SMCB switch provides higher throughput under uniform and nonuniform multicast traffic models than our previously proposed input-based SMCB (I-SMCB) switch, where two inputs share the crosspoint buffers [10].[1]

We show the high throughput under multicast traffic of an O-SMCB switch that uses round-robin selection in its arbitration schemes. We adopt this selection scheme for its simplicity and as an example. Other selection schemes can also be used.

The remainder of this letter is organized as follows. Section II describes the O-SMCB switch model. Section III briefly describes our comparative switch I-SMCB and presents the throughput evaluation of both switches under multicast traffic with uniform and nonuniform distributions. Section IV summarizes our conclusions.

II. Output-based Shared-Memory Crosspoint Buffered (O-SMCB) Switch

To observe the response of the proposed switch under multicast traffic only, the O-SMCB switch is provisioned with one multicast first-in first-out (FIFO) queue at each input. This switch has N^2 crosspoints and $\frac{N^2}{2}$ crosspoint buffers in the crossbar. Figure 1 shows the architecture of the O-SMCB switch. A crosspoint in the buffered crossbar that connects

This work is supported in part by National Science Foundation under Grant Award 0435250.

The authors are with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102. Roberto Rojas-Cessa is the corresponding author. Email: rojas@njit.edu.

[1]We have previously shown that the performance of an SMCB switch is the highest when the number of inputs sharing the buffer is 2 and the lowest when the number is N [10].

input port i to output j is denoted as $CP(i,j)$. The buffer shared by $CP(i,j)$ and $CP(i,j')$ that stores cells for output ports j or j', where $j \neq j'$, is denoted as $SMB(i,q)$, where $0 \leq q \leq \frac{N}{2}-1$. We assume an even N for the sake of clarity. However, an odd N can be used with one input port using dedicated buffers of size 0.5 to 1.0 the size of an SMB. The size of an SMB, in number of cells that can be stored, is k_s. In this letter, we study the case of minimum amount of memory, or when $k_s = 1$ (equivalent to having 50% of the memory in the crossbar of a CICB switch). Therefore, $SMB(i,q)$ with $k_s = 1$ can store a cell that can be directed to either j or j'. The SMB has two egress lines, one per output.

To avoid the need for speedup at SMBs, only one output is allowed to access an SMB at a time. The access to one of the N SMBs by each output is decided by an output-access scheduler. A scheduler performs a match between SMBs and the outputs that share them by using round-robin selection. There are $\frac{N}{2}$ output-access schedulers in the buffered crossbar, one for each pair of outputs. Multicast cells at the inputs have an N-bit multicast bitmap to indicate the destination of the multicast cells. Each bit of the bitmap is denoted as D_j, where $D_j = 1$ if output j is one of the cell destinations, otherwise $D_j = 0$. Each time a multicast copy is forwarded to the SMB for the cell's destination, the corresponding bit in the bitmap is reset. When all bits of a multicast bitmap are zero, the multicast cell is considered completely served. Call splitting is used by this switch to allow effective replication and to alleviate a possible head-of-line blocking.

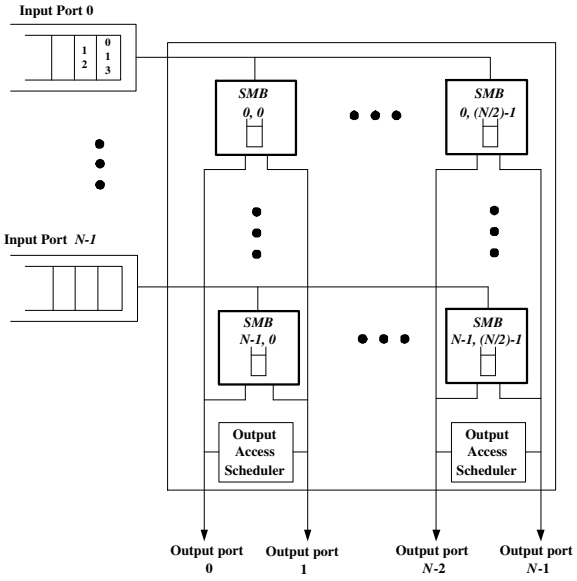

Fig. 1. $N \times N$ O-SMCB switch with shared-memory crosspoints by outputs.

A flow control mechanism is used to notify the inputs about which output replicates a multicast copy and to avoid buffer overflow. The flow control allows the inputs to send a copy of the multicast cell to the crossbar if there is at least one outstanding copy and an available SMB for the destined output. After all copies of the head-of-line multicast cell have been replicated, the input considers that cell served and starts the process with the cell behind.

III. Performance Evaluation

We compare the performance of our proposed O-SMCB switch to those of a CICB and I-SMCB switches. Models of 16×16 O-SMCB, I-SMCB, and CICB switches were implemented in discrete-event simulation programs. Similarly to the O-SMCB, the SMBs are shared in the I-SMCB switch, however, by (two) inputs. Figure 2 shows the I-SMCB switch. For a fair comparison, the I-SMCB also uses round-robin selections for SMB-access by inputs and for output arbitration. The CICB switch uses round-robin for input and output arbitrations. We study the maximum achievable throughput for each switch. The switches were simulated for 500,000 time slots.

We consider multicast traffic models with uniform and nonuniform distributions and Bernoulli arrivals: multicast uniform, multicast diagonal with fanouts of 2 and 4 (called diagonal2 and diagonal4, respectively), and broadcast. In the uniform multicast traffic model, multicast cells are generated with a uniformly distributed fanout among N outputs. For this traffic model, the average fanout is $\frac{1+N}{2} = \frac{17}{2}$ and a maximum admissible input load of $\frac{1}{fanout}$=$\frac{1}{8.5}$. This traffic model includes a fanout=1 or unicast traffic. The multicast diagonal2 traffic model has a destination distribution to $j = i$ and $j = (i+1)\ modulo\ N$ for each multicast cell, and a maximum admissible input load of 0.5. The multicast diagonal4 traffic model has the copies of multicast cells destined to $j = \{i, (i+1)\ modulo\ N, (i+2)\ modulo\ N$, and $(i+3)\ modulo\ N\}$ for each multicast cell, and its admissible input load is 0.25 (i.e., output load=1.0). A broadcast multicast cell generates copies for all N different outputs and has a maximum admissible load is $1/16 = 0.0625$.

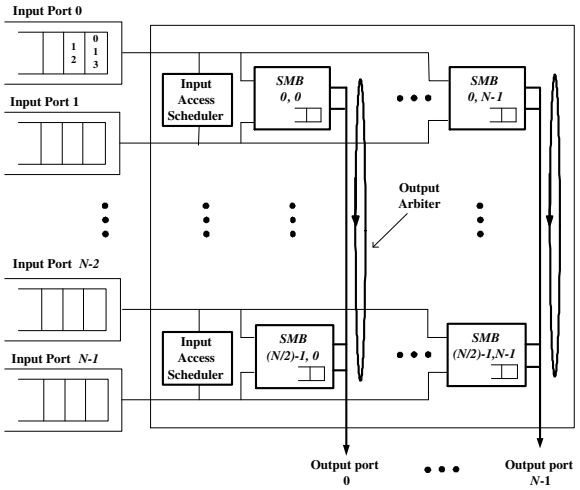

Fig. 2. $N \times N$ I-SMCB switch with shared-memory crosspoints by inputs.

Under admissible multicast uniform traffic, all switches deliver 100% throughput. These results are observed under both Bernoulli and Bursty arrival. Under admissible multicast diagonal2 traffic, the throughputs observed are 100% for the O-SMCB and CICB switches, and 96% for the I-SMCB switch. Under admissible multicast diagonal4 traffic, the performance of the I-SMCB switch decreases to 67%, while the throughputs of the O-SMCB and CICB switches remain close to 100%. Under broadcast traffic (fanout equal to N), the throughput of the O-SMCB switch is 99% and the throughput of the CICB switch is close to 100%, while the throughput of the I-SMCB switch is 95%. The simulation results under diagonal multicast traffic are shown in Figure 3.

Throughput degradation under overload conditions

Multicast is a traffic type difficult to police for admissibility. Furthermore, the performance of switches under inadmissible

traffic (produced by larger fanouts than the expected average) may change. In cases of unicast traffic, the maximum throughput of a switch can remain high with a fair scheduler. However, this might not be the case under multicast traffic. In this experiment, we increased the input load beyond the maximum admissible values in the considered traffic models to observe throughput changes of the O-SMCB, I-SMCB, and CICB switches under these overload conditions. Here, we measured the throughput of the switches as a ratio between the maximum measured throughput and the maximum throughput that a switch is able to provide when all outputs are able to forward a cell.

Under uniform multicast traffic, the throughputs of O-SMCB and I-SMCB switches degrade to 93% when the input load is larger than 0.117 (i.e., output load is larger than 1.0), while the throughput of the CICB switch is 100%. This throughput degradation in the SMCB switches occurs because of the increased number of contentions for SMB access as the traffic load increases. Under multicast diagonal2 traffic, the throughputs of the I-SMCB and CICB switch drop to 96% and 93%, respectively, while the throughput of the O-SMCB switch remains close to 100%. Under multicast diagonal4 traffic, the throughput of the I-SMCB switch drops to 68%, while the throughputs of the O-SMCB and CICB switches remain close to 100%. Under broadcast traffic, the throughput of the I-SMCB switch decreases to 79%. However, the throughputs of the O-SMCB and CICB switches remain close to 100%.

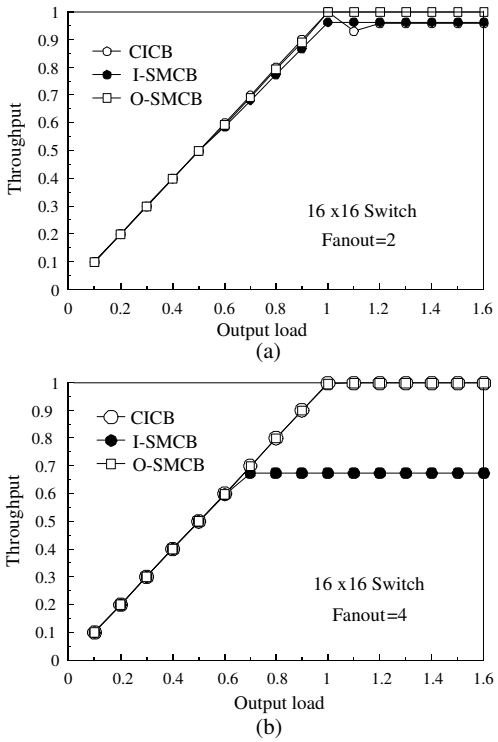

Fig. 3. Throughput performance of 16×16 I-SMCB and O-SMCB switches under a) diagonal2 traffic and b) diagonal4 traffic.

Table I summarizes the obtained throughput for all tested traffic models. In this table, Ta stands for the measured throughput under admissible traffic and Ti for the measured throughput under inadmissible traffic. The letters I, O, and C in parenthesis indicate that a result is related to the I-SMCB, O-SMCB, and CICB switches, respectively. As seen in this table, the performance of the O-SMCB switch is comparable to that of the CICB switch and higher than that of an I-SMCB switch. Therefore, the O-SMCB switch provides comparable performance but with 50% the memory amount of a CICB switch.

TABLE I
THROUGHPUT UNDER MULTICAST TRAFFIC.

Traffic type	$Ta(I)$	$Ta(O)$	$Ta(C)$	$Ti(I)$	$Ti(O)$	$Ti(C)$
Uniform	100%	100%	100%	93%	93%	100%
Diagonal2	96%	100%	100%	96%	100%	93%
Diagonal4	67%	100%	100%	68%	100%	100%
Broadcast	95%	99%	100%	79%	100%	100%

IV. CONCLUSIONS

Here, we proposed a novel switch architecture to support multicast traffic using a shared-memory switch that shares crosspoint buffers among outputs to use 50% of the memory amount in the crossbar fabric that CICB switches require. Our proposed switch, the O-SMCB switch, delivers high performance under multicast traffic while using no speedup. Furthermore, the proposed switch shows an improved performance over our previously proposed switch with shared memory among inputs. The improved switch is based on having the buffers shared by the outputs instead of the inputs. This has the effect of facilitating call splitting by allowing inputs directly access the crosspoint buffers. This simple improvement has a significant impact on switching performance. As a result, the O-SMCB provides 100% throughput under both admissible uniform and diagonal multicast traffic with fanouts of 2 and 4. Furthermore, our proposed switch keeps the throughput high under nonuniform traffic with overloading conditions. The disadvantage of SMCB switches is that time relaxation of CICB switches is minimized because of the matching process used in buffer access. However, the matching is performed in chip and among a moderate number of outputs. Furthermore, the matching process is simpler in the SMCB switches than those used in IB switches for multicast traffic. The O-SMCB switch, with buffer space for $\frac{N^2}{2}$ cells, provides comparable performance to that of a CICB switch, with buffer space for N^2 cells, therefore, saving 50% of the amount of memory.

REFERENCES

[1] T.T. Lee, "Non-blocking copy networks for multicast packet switching," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 1455-1467, December 1988.

[2] M. Karol, M. Hluchyj, "Queuing in High-performance Packet-switching," *IEEE J. Select. Areas Commun.*, Vol. 6, pp. 1587-1597, December 1988.

[3] Y. Doi and N. Yamanaka, "A High-Speed ATM Switch with Input and Cross-Point Buffers," *IEICE Trans. Commun.*, Vol. E76, No.3, pp. 310-314, March 1993.

[4] E. Oki, N. Yamanaka, Y. Ohtomo, K. Okazaki, and R. Kawano, "A 10-Gb/s (1.25 Gb/s x8) 4 x 0.25-μm CMOS/SIMOX ATM Switch Based on Scalable Distributed Arbitration," *IEEE J. Solid-State Circuits*, Vol. 34, No. 12, pp. 1921-1934, December 1999.

[5] M. Nabeshima, "Performance Evaluation of a Combined Input- and Crosspoint-Queued Switch," *IEICE Trans. Commun.*, Vol. E83-B, No. 3, pp. 737-741, March 2000.

[6] K. Yoshigoe and K.J. Christensen, "A parallel-polled Virtual Output Queue with a Buffered Crossbar," *Proceedings of IEEE HPSR 2001*, pp. 271-275, May 2001.

[7] R. Rojas-Cessa, E. Oki, Z. Jing, and H. J. Chao, "CIXB-1: Combined Input-One-Cell-Crosspoint Buffered Switch," *Proc. IEEE Workshop on High Performance Switches and Routers 2001*, pp. 324-329, May 2001.

[8] M. Hamdi and M. Lhamdi, "Scheduling multicast traffic in internally buffered crossbar switches," *Proc. IEEE International Conference on Communications 2004*, Vol. 2, pp. 1103-1107, June 2004.

[9] S. Sun, S. He, Y. Zheng, and W. Gao, "Multicast scheduling in buffered crossbar switches with multiple input queues," *Proc. IEEE Workshop on High Performance Switches and Routers 2005*, 12-14, pp. 73-77, May 2005.

[10] Z. Dong and R. Rojas-Cessa, "Long Round-Trip Time Support with Shared-Memory Crosspoint Buffered Packet Switch," *Proc. IEEE Hot Interconnects 2005*, pp. 138-143, August 2005.