On the Combined Input-Crosspoint Buffered Switch with Round-Robin Arbitration

Roberto Rojas-Cessa, Eiji Oki, and H. Jonathan Chao

***Abstract*— Input-buffered switches have been widely considered for implementing feasible packet switches. However, their matching process may not be time efficient for switches with high-speed ports. Buffered crossbars are an alternative to relax timing for packet switches with high-speed ports and to provide high-performance switching. Buffered crossbar switches were originally considered expensive as the memory amount required in the crosspoints is proportional to the square of the number of ports ($O(N^2)$). This limitation is now less stringent with the advances on chip fabrication techniques and when considering small crosspoint buffer sizes. In this paper, we study a combined input-crosspoint buffered packet switch, named CIXB, with Virtual Output Queues (VOQs) at the inputs, and arbitration based on round-robin selection. We show that the CIXB switch achieves 100% throughput under uniform traffic and high performance under nonuniform traffic, using one-cell crosspoint buffer size and no speedup.**

***Index Terms*— Buffered crossbar, crosspoint-buffered switch, round-robin arbitration, virtual output queue, credit-based flow control.**

I. Introduction

As interconnection technologies mature and new multimedia coding techniques emerge, networks are required to allow a large number of Internet traffic flows through. Therefore, the search for switch/router architectures with high performance and large capacity persists to satisfy this demand.

It is common to find the following practices in packet switch design. 1) Segmentation of incoming variable-size packets at the ingress side of a switch to perform internal switching with fixed-size packets, or cells, and re-assembling the packets at the egress side before they depart from the switch. 2) Use of separate queues at the inputs, one for each output, known as virtual output queues (VOQs) to avoid head-of-line (HOL) blocking [1]. 3) Use of crossbar fabrics for implementation of packet switches because of their non-blocking capability, simplicity, and market availability. This paper follows these practices.

In general, the performance of packet switches are constrained by memory speed and the efficiency of the arbitration scheme used to select the cells that will traverse the switch at a given time. Output Buffered (OB) switches use no arbitration scheme at the inputs as the output buffers run at N times faster than the speed of external links. Therefore, for a limited memory speed, the size of such switches is restricted to a small number of ports as link rates increase. On the other hand, the memory in Input Buffered (IB) switches runs at link-rate speeds, making these switches attractive. However, IB switches need to perform matching between input and outputs to resolve input and output contentions. Fast matching schemes for IB switches can be modeled as parallel matchings, where N input and output arbiters perform the parallel selection and communication among them to decide the matching results. This communication adds overhead time to the matching process. The parallel matching process can be characterized by three phases: request, grant, and accept [5]. Therefore, the resolution time would be the time spent in each of the selection phases plus the transmission delays for the exchange of request, grant, and accept information. As an example, a matching scheme must perform input or output arbitration within 6.4 ns in an IQ switch with 40 Gbps (OC-768) ports and 64-byte cells, as input and output arbitrations may use up to half of a time slot, assuming that the transmission delays are decreased to a negligible value (e.g., the arbiters are implemented in a single chip, in a centralized way).

A solution to minimize the time overhead is to use buffers in the crosspoints of a crossbar fabric, or buffered crossbar. In a buffered crossbar switch, an input can avoid waiting for a matching to be completed before a cell is forwarded to the crossbar. However, the number of buffers grows in the same order as the number of crosspoints ($O(N^2)$), making implementation costly or infeasible for a large buffer size or a large number of ports. As VLSI technology has matured, buffered crossbars can be considered feasible. An example of a buffered crossbar was proposed in [11], where a 2×2 crossbar chip with a crosspoint memory of 16 kbytes was implemented

Partial results of this paper were published in the Proceedings of the IEEE 2001 Workshop on High Performance Switches and Routers, May 2001, Dallas, Texas.

Roberto Rojas-Cessa is the correspondence author. He is with the Dept. of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102. Email: rrojas@njit.edu. This work is partially sponsored by NSF Awards 0435250 and 0423305.

Eiji Oki is with NTT Network Service Systems Laboratories, Tokyo, 180-8585 Japan, Email: oki.eiji@lab.ntt.co.jp.

H. Jonathan Chao is with the Dept. of Computer and Electrical Engineering, Polytechnic University, Brooklyn, New York 11201. Email:chao@poly.edu. This work is partially sponsored by NSF Award 0435303.

to provide an acceptable cell loss.

Placement of input buffers, used for reducing the memory amount at the crosspoints at a given cell loss rate, gives place to combined input-crosspoint buffered (CICB) switches. Examples of CICB switches with single-cell crosspoints were proposed [12]-[15]. The switches have FIFO input buffers at the input ports. The FIFOs limit the maximum achievable throughput of this switch because of HOL blocking becomes present. As with maximal-matching schemes in non-buffered crossbars, the HOL blocking for FIFO buffers can be overcome in a buffered crossbar with VOQs. In [19] and [20], buffered crossbars with VOQs at the inputs were studied. The switch in [19] is targeted for traffic with service guarantees. The switch uses a minimum crosspoint buffer size of two internally-sized packets, speedup and output buffers. In [20], the switch uses a cell-based switching, and it addresses guaranteed and best effort traffic. This architecture needs a large buffer crosspoint size for overflow avoidance as it uses a back-pressure flow control [22], as we show in Section III. Furthermore, it has been shown that CICB switches can emulate OB switches with a speedup of 2 [26], [27], with a complex queuing policy. However, output emulation is out of the scope of this paper. Our objective is to study the throughput achieved by round-robin based arbitration schemes, without speedup, and under admissible traffic patterns.

Our motivation is to find the minimum sufficient buffer size in a crosspoint such that no speedup is used to provide high switching performance, and to study further the impact of this buffer size under different traffic types.

In this paper, we study the use of one-cell crosspoint buffers in a CICB packet switch with VOQs at the inputs and with simple round-robin arbitration schemes for input and output arbitrations. We show that a CICB switch with round-robin selection, named CIXB, provides 100% throughput for uniform traffic and high performance under several nonuniform admissible traffic patterns. We also show that CIXB uses the simplest arbitration scheme that produces satisfactory switching performance among all round-robin schemes considered in this paper.

We show that the performance of the proposed switch is optimum for uniform traffic by studying the effect of the crosspoint-buffer size, and by comparing the performance of CIXB to that of an OB switch. In a CICB switch, input and output arbitrations are performed separately, and, therefore, this switch provides a solution for high-speed port switches. Following the example above, a port with 40-Gbps speed and 64-byte cells, would provide up to 12.8 ns (or the complete cell slot) for input or output arbitration and thus, relaxing the arbitration time. Furthermore, the memory in a CICB switch runs at the same speed as in an IB switch.

This paper is organized as follows. Section II, we presents the CIXB switch model. Section III discusses the properties and analysis of CIXB with one-cell crosspoint buffers. Section IV presents a performance study of this switch under uniform and non-uniform traffic patterns. Section V describes the switch model for non-negligible round-trip times. Section VI presents our conclusions.

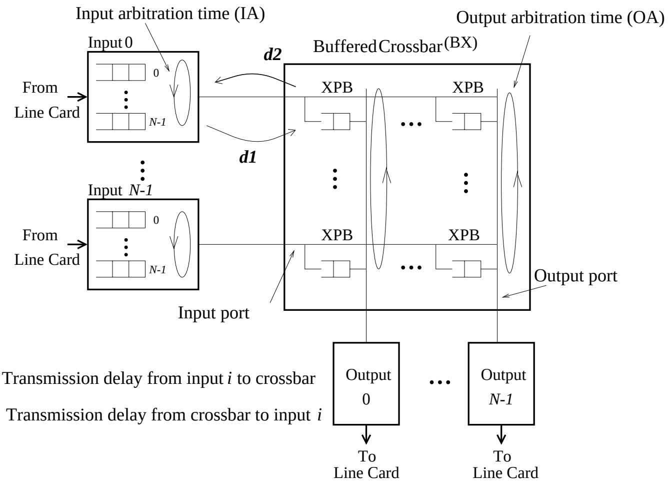

Fig. 1. CIXB switch model.

II. Combined Input-Crosspoint Buffered (CICB) Switch Model

A buffered crossbar (BX) has N input and output ports. A crosspoint (XP) element in the BX that connects input port i, where $0 \leq i \leq N-1$, to output port j, where $0 \leq j \leq N-1$, is denoted as $XP_{i,j}$. Figure 1 shows the architecture of a CICB switch. The switch has the following major components:

Input Buffers. There are N VOQs at each input. A VOQ at input i that stores cells for output j is denoted as $VOQ_{i,j}$.

Crosspoint Buffer (XPB). The XPB of $XP_{i,j}$ is denoted as $XPB_{i,j}$. The size of $XPB_{i,j}$ is k cells. We call a CIXB switch with an XPB size k, CIXB-k.

Flow Control. A credit-based flow-control mechanism indicates input i whether $XPB_{i,j}$ has room available for a cell or not. Each VOQ has a credit counter, where the maximum count is the number of cells that $XPB_{i,j}$ can hold. When the number of cells sent by $VOQ_{i,j}$ reaches the maximum count, $VOQ_{i,j}$ is considered not eligible for input arbitration and overflow on $XPB_{i,j}$ is avoided. The count is increased by one every time a cell is sent to $XPB_{i,j}$, and decreased by one every time that $XPB_{i,j}$ forwards a cell to output j. If $XPB_{i,j}$ has room for, at least, one cell, then $VOQ_{i,j}$ is considered eligible by the input arbiter.

Input and output arbitration. Round-robin arbitration is used at the input and output ports. An input arbiter selects an eligible $VOQ_{i,j}$ for sending a cell to $XPB(i,j)$. VOQ elegibility is determined by the flow-control mechanism. An output arbiter selects a non-empty $XPB_{i,j}$ for forwarding. Input and output arbitrations are performed separately, therefore, reducing cell selection complexity.

III. Properties of CIXB with One-Cell Crosspoint Buffers

In this section, we refer CIXB-1 and assume the following traffic model:

All inputs have uniform traffic; where each $VOQ_{i,j}$ receives cells at rate $\rho_{i,j}$ such that $\sum_{i=0}^{N-1} \rho_{i,j} = 1.0$, $\sum_{j=0}^{N-1} \rho_{i,j} = 1.0$, and each $VOQ_{i,j}$ has an occupancy $\sigma_{i,j}(t)$, such that $0 < \sigma_{i,j}(t) < \infty$ and $\sigma_{i,j}(t)$ is large. [1]

CIXB-1 reaches a state such that the number of cells entering the system equals the number of cells leaving from it, defining 100% throughput. CIXB-1 has the following property:

Property 1: CIXB-1 achieves 100% throughput under uniform traffic.

The proof of this property is shown in the Appendix. For this property to be effective, we need to consider the round-trip time. We define round-trip time (RT) as the sum of the delays of the input arbitration (IA), the transmission of a cell from an input port to the crossbar ($d1$), the output arbitration (OA), and the transmission of the flow-control information back from the crossbar to the input port ($d2$). Figure 1 shows an example of RT for input 0 by showing the transmission delays for $d1$ and $d2$, and arbitration times, IA and OA. Cell and data alignments are included in the transmission times. The condition for CIXB-1 to provide 100% throughput is such that, in general:

$$RT = d1 + OA + d2 + IA \leq k. \qquad (1)$$

In other words, RT time slots must be absorbed by the number of available cells in the XPB. The arbitration times IA and OA are constrained to $IA \leq 1$ and $OA \leq 1$.

As the cost of implementing memory in a chip is still considerable, although feasible with currently available VLSI technologies, it is important to minimize the XPB size within the implementation limits. A back-pressure flow control is expensive to consider with CIXB as the XPB size needs to be at least twice RT ($k \geq 2RT$) to avoid underflow and, therefore, performance degradation. To evaluate this, consider the worst-case scenario: there is only a single flow in BX from input i to output j. In this case, $XPB_{i,j}$ must have room available for all cells that can be transmitted to their outputs while the notification of back-pressure release travels back to input i and a cell travels from input i to $XPB_{i,j}$ (RT time), so that output port j can continue forwarding cells from input port i. This is why the use of a credit-based flow control is more cost-effective in CIXB.

IV. Throughput and Delay Performance

We studied the performance of an IB switch using iSLIP scheduling, two combined input-crosspoint buffered switches, and an OB switch by computer simulation. One of the CICB switches is the CIXB-1 and the other uses a pre-determined permutation for input and output selections, where the permutation changes cyclically and in a fixed sequence, as time division multiplexing (TDM) works. TDM also uses one-cell XPB. We consider iSLIP as the scheduling scheme in the IB switch, as iSLIP is based on round-robin selection. We show a comparison on the performance between IB and CICB switches, where round-robin selection is used. The comparison with the OB switch is used the best possible performance. The traffic patterns studied in this section are uniform and non-uniform traffic with Bernoulli arrivals. We also consider traffic with uniform distribution with bursty arrivals. In our switch model, we do not consider segmentation and re-assembly delays. Our simulation results are obtained with a 95% confidence interval, not greater than 5% for the average cell delay.

A. Uniform Traffic

We study an IB switch with 1SLIP and 4SLIP, CIXB-1, TDM, and OB under uniform traffic with Bernoulli and bursty arrivals (on-off modulated markov process) for a switch size of 32×32. Note that 4 is the optimum number of iterations for iSLIP, as more iterations offer no measurable improvements [5]. Figure 2 shows that, under this traffic model, all switches provide 100% throughput. CIXB-1 provides an average delay close to OB. 1SLIP and TDM have similar average delay for high loads, and CIXB-1 delivers shorter average cell delay than 4SLIP and TDM. The reason why CIXB delivers this high performance is because at a given time slot, there could be more than one cell from input i stored in crosspoints for different outputs that depart at the same time. Therefore, several cells from input i could leave the switch in one time slot. In an IB switch, the number of cells that can traverse to an output port is one.

This small delay of CIXB-1 remains independently of the switch size according to our preliminary results [21]. This low average cell delay is expected as it is similar to that of an OB switch.

Figure 3 shows the average delay performance of TDM, CIXB-1, and OB under on-off arrival uniform traffic. The traffic in this figure has an average burst length (l) of 1, 10, and 100 cells, where the burst length is exponentially distributed. The figure shows that CIXB-1 follows the throughput and average cell delay of an OB switch. TDM has a significantly larger delay than CIXB-1. Some differences can be noted between CIXB-1 and OB when the input load is close to 1.0.

[1] For simplicity, we assume initial condition in the demonstration of this property; however, we can start from an empty buffer as well. This condition is justified in the Appendix.

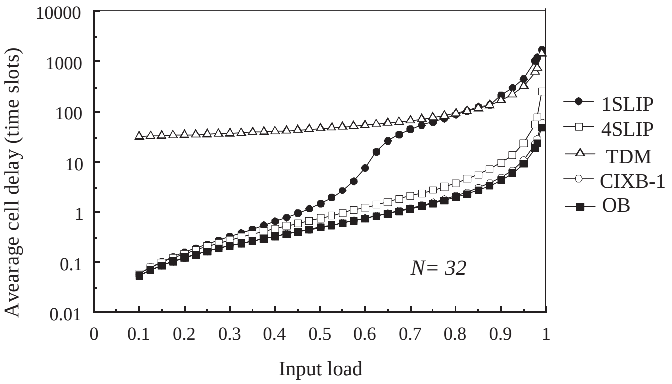

Fig. 2. Performance of CIXB, TDM, and OB under uniform traffic.

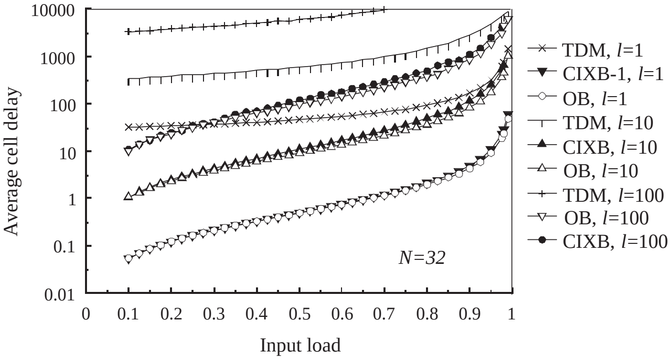

Fig. 3. Performance of CIXB, TDM, and OB under bursty uniform traffic.

B. Nonuniform Traffic

We simulated iSLIP, TDM, and CIXB under three traffic models with Bernoulli arrivals and nonuniform distributions: unbalanced [21], asymmetric [25], and Chang's models [24].

1) Unbalanced Traffic: The unbalanced traffic model uses a probability w as the fraction of input load directed to a single predetermined output, while the remaining load is directed to all outputs with a uniform distribution. Let us consider input port s, output port d, and the offered input load for each input port ρ. The traffic load from input port s to output port d, $\rho_{s,d}$ is given by $\rho_{s,d} = \rho\left(w + \frac{1-w}{N}\right)$ if $s = d$, and $\rho_{s,d} = \rho\frac{1-w}{N}$ otherwise, where N is the switch size. Here, the aggregate offered load that goes to output d from all input ports, ρ_d is given by $\rho_d = \sum_s \rho_{s,d} = \rho\left(w + N \times \frac{1-w}{N}\right) = \rho$. When $w = 0$, the offered traffic is uniform. On the other hand, when $w = 1$, the traffic is completely directional. This means that all the traffic of input port s is destined for output port d only, where $s = d$. Figure 4 shows iSLIP, CIXB-1 and TDM under unbalanced traffic for $0 \leq w \leq 1$, and observed the minimum throughput in terms of w achieved by each switch. The throughput of iSLIP with 1 and 4 iterations are 64% and 80%. The throughput of CIXB-1 is 84%, higher than iSLIP. The throughput for TDM drops to almost 0 when w =1.0, because of the predetermined connectivity that TDM provides, independently of the existing traffic. CIXB provides a higher throughput than IB switches with round-robin schemes [21] at the expense of having buffers in the crosspoints. These results clearly show the advantage of using a round-robin scheme and crosspoint buffers under unbalanced traffic.

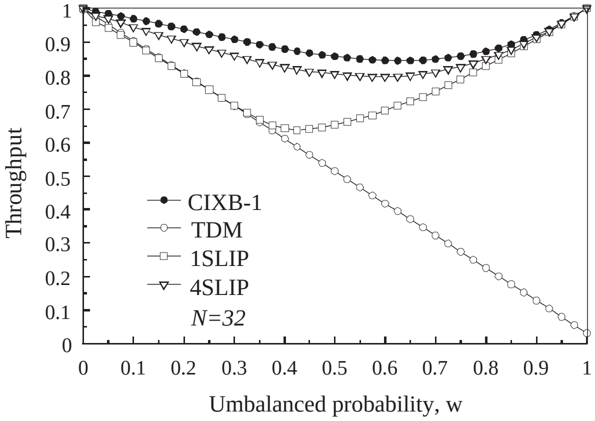

Fig. 4. Througput of iSLIP, TDM, and CIXB-1 under unbalanced traffic.

2) Chang's and Asymmetric Traffic: Chang's traffic model can be defined as $\rho = 0$ for $i = j$ and $\rho = \frac{1}{N-1}$, otherwise. The asymmetric traffic model can be defined as having different load for each input-output pair, such as $\rho_{i,(i+j) \bmod N} = a_j\rho$, where $a_0 = 0$, $a_1 = (r-1)/(r^N-1)$, $a_j = a_1 r^{j-1}$ $\forall j \neq 0$, and $\rho_{i,j}/\rho_{(i+1) \bmod N}$, $j = r$, $\forall i \neq 0$, $(i+1) \bmod N \neq 0$, and $r = (100 : 1)^{-1/(N-2)}$. Figure 5 shows the average cell delay of iSLIP, TDM, and CIXB-1 under asymmetric and Chang's traffic models. 1SLIP delivers up to 75% throughput under asymmetric traffic and close to 99% throughput under Chang's traffic. 4SLIP delivers close to 100% throughput under these two traffic patterns. TDM provides 20% throughput under asymmetric traffic, and close to 99% throughput under Chang's traffic. CIXB delivers 100% throughput and lower average cell delay than the other switches under both traffic models. These results show that having buffers in the crosspoints and round-robin arbitration improves the switching performance.

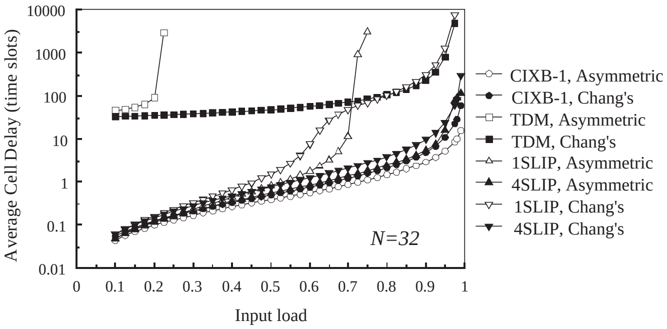

Fig. 5. Delay performance of TDM and CIXB-1 under Asymmetric and Chang's traffic models.

C. Buffer and Switch Size Effect on CIXB-k

We simulated a 32×32 CIXB-k for $k = \{1, 2, 8\}$ under uniform traffic and observed the average delay performance and tail delay distribution. Because the similarity of the average cell delay in CIXB-1 and in OB switches, the value of k in CIXB-k produces non-measurable differences on the average delay performance under Bernoulli and bursty ($l = 10$) uniform traffic, as shown previously [21].

A more stringent test is performed by measuring the tail delay distribution of CIXB-k. Figure 6 shows that the tail delay distribution under Bernoulli uniform traffic, for different input loads, presents non-measurable differences for any k values. This is because as long as the k value complies with Eq. (1), the crosspoint buffers avoid underflow states. By providing a larger k, the number of cells leaving BX will not be larger as the optimum performance of the switch has already been reached. Therefore, the crosspoint buffer size can be kept as small as possible to minimize the amount of in-chip memory without having performance degradation.

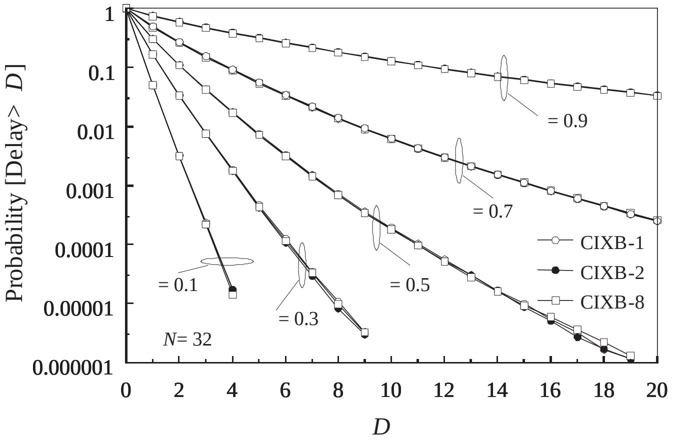

Fig. 6. Tail delay distribution for different input loads and k under uniform traffic.

The effect of k under non-uniform traffic is studied in a 32×32 CIXB-k, with k values between 1 and 32. Figure 7 shows the effect of k in CIXB under unbalanced traffic. This figure shows that the throughput of CIXB under non-uniform traffic is slightly improved when a larger k is provided. When $k = N = 32$, the throughput barely reaches 99%. These results indicate that to achieve 100% throughput under this traffic pattern, a very large k is needed. To provide a higher throughput under unbalanced traffic, a weighted round-robin scheme [5], [23] can be used. The weight of an input or output can be assigned by considering the queue occupancy or cell ages.

According to previous results, the high performance of CIXB is also independent of the switch size [21]. This result differs from the one that has been observed in IB switches.

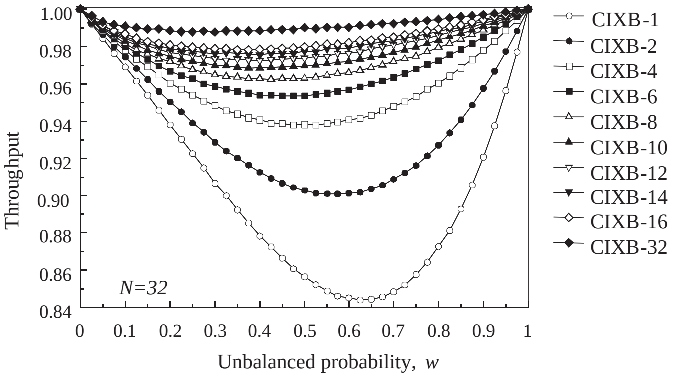

Fig. 7. Effect of buffer size under unbalanced traffic for CIXB-k.

V. Non-negligible Round-Trip Times and CIXB-$(k - RT)$

In the cases above, we assumed small round-trip times, such that Eq. (1) is satisfied with $k = 1$. Although the consideration of $k = 1$ is practical, the distance between the buffered crossbar and the port cards may be longer than one time slot. Therefore, the k needs to be increased to comply with Eq. (1).

In general, CIXB can be denoted as CIXB-$(k - RT)$, where $k \geq RT$, for non-negligible RT values (i.e., $RT \geq 1$). We call this the general CIXB model. To observe the performance of a CIXB-($k - RT$) switch, we simulated a switch with different k and RT values for $k - RT = 0$ under unbalanced traffic. Figure 8 shows the throughput performance of CIXB-($k - RT$). The throughput performance is improved by larger k values and slightly decreased by RT. The improvement, however, seems to be of greater effect.

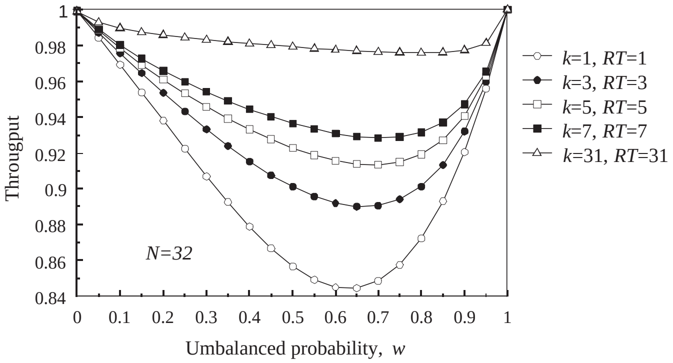

Fig. 8. Effect of buffer size and RT under unbalanced traffic in CIXB-$(k - RT)$.

VI. Conclusions

We presented a study of a Combined Input-One-Cell-Crosspoint Buffered (CIXB-1) switch model. We showed that each crosspoint buffer with capacity for one cell is enough to provide 100% throughput under uniform traffic. CIXB-1 offers not only better timing properties over IB switches with an arbitration scheme, such as iSLIP, but a higher performance as CIXB-1 is able to forward more than one cell from an input to several outputs at a given time slot. Our simulation shows that CIXB-1 delivers an close average delay equivalent to that of an output buffered switch under uniform traffic with Bernoulli and bursty arrivals. CIXB presents high throughput and timing relaxation that allow high-speed arbitration and scalability. The timing relaxation eases the design of cell alignment at the input ports and allows the use of a complete time slot for arbitration.

A crossbar with a single-cell crosspoint buffer is not limited by the amount of memory. The pin count remains the major limitation as in the case for non-buffered crossbars switches. The memory amount disadvantage of a crossbar is not a bottleneck for a small buffer size as in CIXB-1 and medium-sized switch modules. As an example, a 128×128 CIXB-1 switch module requires 8 Mb for 64-byte cells, while more than 32-Mb memory can be embedded in a chip using available 0.11-μm ASIC CMOS technology. Our previous results show that the delay performance of CIXB is independent of the crosspoint-buffer size k and switch size under Bernoulli uniform traffic. Under unbalanced traffic, the throughput of CIXB-k approaches 100% slowly and asymptotically as the k value increases. We extended the CIXB-k model to a general model, CIXB-($k - RT$), by taking into account the round-trip times. An interesting future work is to find arbitration schemes that provide performance guarantees with no speedup.

References

[1] M. Karol, M. Hluchyj, “Queuing in High-performance Packet-switching,” *IEEE J. Select. Area Commun.*, vol. 6, pp. 1587-1597, December 1988.

[2] T. Anderson, S. Owicki, J. Saxe, and C. Thacker, “High speed switch scheduling for local area networks,” ACM Trans. Comput. Syst., vol. 11, no. 4, pp. 319-352, November 1993.

[3] N. McKeown, A. Mekkittikul, V. Anantharam, J. Walrand, “Achieving 100% Throughput in an Input-queued Switch,” *IEEE Trans. Commun.*, vol. 47, no. 8, pp. 1260-1267, August 1999.

[4] A. Mekkittikul, N. McKeown, “A Practical Scheduling Algorithm to Achieve 100% throughput in input-queued switches,” in Proc. *INFOCOM ’98*, vol. 2 , pp. 792 -799, March 1998.

[5] N. McKeown, “The iSLIP Scheduling Algorithm for Input-Queue Switches,” *IEEE/ACM Trans. Networking*, vol. 7, no. 2, pp. 188-200, April 1999.

[6] P. Krishna, P., Patel, N. S., Charny, A., Simcoe, R., “On the Speedup Required for Work-Conserving Crossbar Switches,” *IEEE J. Select. Areas Commun.*, vol. 27, no. 6, pp. 1052-1066, June 1999.

[7] G. Nong, J. K. Muppala, M. Hamdi, “Analysis of nonblocking ATM switches with multiple input queues,” *IEEE/ACM Trans. Networking*, vol. 7, issue 1 , pp. 60-74, February 1999.

[8] G. Nong, M. Hamdi, J.K. Muppala, “Performance evaluation of multiple input-queued ATM switches with PIM scheduling under bursty traffic,” *IEEE Trans. Commun.* , Vol. 49, issue 8 , pp. 1329 -1333, August 2001.

[9] N. McKeown, M. Izzard, A. Mekkittikul, W. Ellersick, M. Horowitz, “Tiny Tera: A Packet Switch Core,” *IEEE Micro*, vol. 17, no. 1, pp. 26 -33, January-February 1997.

[10] Texas Instruments, GS40 0.11-μm CMOS, Standard Cell/Gate Array, http://www.ti.com/, version 1.0, January 2001.

[11] S. Nojima, E. Tsutsui, H. Fukuda, and M. Hashimmoto, “Integrated Packet Network Using Bus Matrix,” *IEEE J. Select. Areas Commun.*, vol. SAC-5, no. 8, pp. 1284-1291, October 1987.

[12] A. K. Gupta, L. O. Barbosa, and N. D. Georganas, “16 x 16 Limited Intermediate Buffer Switch Module for ATM Networks,” in Proc. *GLOBECOM ’91*, pp. 939-943, December 1991.

[13] A. K. Gupta, L. O. Barbosa, and N. D. Georganas, “Limited Intermediate Buffer Switch Modules and their Interconnection Networks for B-ISDN,” in Proc.*ICC ’92*, pp. 1646-1650, June 1992.

[14] 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.

[15] 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.

[16] F. A. Tobagi, “Fast Packet Switch Architectures For Broadband Integrated Services Digital Networks,” *Proceedings of the IEEE*, vol. 78, no. 1, pp. 133-167, January 1990.

[17] H. Ahmadi and W. E. Denzel, “A Survey of Modern High-Performance Switching Techniques,” *IEEE J. Select. Areas Commun.*, vol. 7, no. 7, pp. 1091-1103, September 1989.

[18] R. Y. Awdeh and H. T. Mouftah, “Survey of ATM Switch Architectures,” *Computer Networks and ISDN Systems*, vol. 27, pp. pp. 1567-1613, November 1995.

[19] D. C. Stephen and H. Zhang, “Implementing Distributed packet Fair Queuing in a Scalable Switch Architecture,” in Proc. *INFOCOM ’98*, vol. 1, pp. 282-290, 1998.

[20] F. M. Chiussi and A. Francini, “A Distributed Scheduling Architecture for Scalable Packet Switches,” *IEEE J. Select. Areas Commun.*, pp. 2665-2683, December 2000.

[21] R. Rojas-Cessa, E. Oki, Z. Jing, and H. Jonathan Chao, “CIXB-1: Combined Input-One-cell-crosspoint Buffered Switch,” in Proc. *IEEE HPSR 2001*, pp. 324-329, May 2001.

[22] F. M. Chiussi, Y. Xia, and V.P. Kumar, “Backpressure in shared-memory based atm switches under multiplexed bursty sources,” *IEEE INFOCOM’96*, pp. 830-843, March 1996.

[23] B. Li, M. Hamdi, X-R. Cao, “An efficient scheduling algorithm for input-queuing ATM switches,” in Proc. *IEEE Broadband Switching Systems Proceedings, 1997*, pp. 148-154, 1997.

[24] C-S. Chang, D-S. Lee, and Y-S. Jou, “Load Balanced Birkhoff-von Neumann Switches,” in Proc. *IEEE HPSR 2001*, pp. 276-280, May 2001.

[25] R. Schoenen, G. Post, and G. Sander, “Weighted Arbitration Algorithms with Priorities for Input-Queued Switches with 100% Throughput,” in Proc. *Broadband Switching Symposium’99*, 1999. http://www.schoenen-service.de/assets/papers/Schoenen99bssw.pdf

[26] L. Mhamdi, M. Hamdi, “Output queued switch emulation by a one-cell-internally buffered crossbar switch Mhamdi,” in Proc. *IEEE Global Telecommunications Conference*, 2003, Volume: 7 , pp. 3688-3693, Dec. 2003.

[27] R. B. Magill, C. E. Rohrs, R. L. Stevenson, “Output-queued switch emulation by fabrics with limited memory,” IEEE JSAC, Vol. 21 , Issue: 4, pp. 606-615, May 2003.

Appendix

In this section we prove Property 1. The switch and traffic model described in Section III are considered for this proof.

Definition 1. $VOQ_{i,j}$ is active if it has at least one buffered cell.

Definition 2. $VOQ_{i,j}$ is inhibited to send a cell to BX when $XP_{i,j}$ is occupied.

Definition 3. Input port i is totally inhibited when all VOQs in it are inhibited.

Definition 4. Output j is active at time slot t if it has a cell to be delivered at time slot t.

Definition 5. Output port j is a non-active output at time slot t if it has no cell to be delivered at time slot t.

Let us denote $X(t)$ to the number of inhibited input ports. $XP_{i,j}(t)$ is the state of crosspoint buffer $XPB_{i,j}$ at the beginning of time slot t. If a previous cell that was not granted by the output arbiter remains in the crosspoint, $XP_{i,j}(t) = 1$. Otherwise, $XP_{i,j}(t) = 0$.

Using the definitions presented above, 100% throughput is re-phrased as: *100% throughput is achieved when $X(t) = 0$ and all VOQs are served with an average rate equal to the average arrival rate.*

Theorem 1: CIXB-1 achieves 100% throughput under uniform traffic.

We prove Theorem 1, by proving Lemmas 1 and 2.

Lemma 1. In CIXB-1, no input is inhibited after N time slots from any initial condition.[2] $X(t) = 0$.

Proof of Lemma 1. We use the following facts:

Fact 1. An input can send at most one cell to BX per time slot.

Fact 2. At output port j, occupied crosspoint $XP_{i,j}$, where $0 \leq i \leq N-1$, is served within N time slots.

We prove this lemma by contradiction. Let us have the initial time $t = 0$. Assume that input i is inhibited at time $t = N$. All the crosspoints related to input i are in state $XP_{i,j}(N) = 1$, such that $\sum_j XP_{i,j}(N) = N$.

Independently of the initial state at $t = 0$,[3] these N cells could have been issued by input i only during the previous $N-1$ time slots ($1 \leq t \leq N-1$) due to Fact 2. However, according to Fact 1, during the previous $N-1$ time slots ($1 \leq t \leq N-1$) an input could send at most $N-1$ cells.

This contradicts the assumption of $\sum_j XP_{i,j}(N) = N$. Therefore, $\sum_j XP_{i,j}(N) \leq N-1$. In the same way, $\sum_j XP_{i,j}(t) \leq N-1$ for any $t > N$. After N time slots from the initial time, there is at least one $XP_{i,j}$ such that $XP_{i,j}(t) = 0$. Therefore, $X(t) = 0$ is kept.

□

Lemma 2. After N time slots of the initial condition, each input port sends a cell from a different VOQ at each time slot in an N-slot cycle.

We can re-phrase Lemma 2.

$VOQ_{i,j}$ has sent one and only one cell in an N-slot time cycle after N time slots of the initial condition (such that a different $VOQ_{i,j}$ sends a cell every time slot).

We consider the remaining facts of CIXB-1:

Fact 3. An output port forwards at most one cell (or grant a single request) each time slot.

Fact 4. At input i, any available crosspoint $XP_{i,j}$, where $0 \leq j \leq N-1$, receives a cell within N time slots.

Definition 6. In input i that uses round-robin arbitration, there are two VOQs, $VOQ_{i,j}$ and $VOQ_{i,j'}$ where $j \neq j'$, such that $VOQ_{i,j}$ is called VOQ^c if this VOQ sends two cells within N time slots, and $VOQ_{i,j'}$ is called VOQ^e if this VOQ does not send a cell within N time slots. These terms are assigned to the VOQs at the time slot where any of these conditions becomes true.

Definition 7. There are two possible relationships between VOQ^c and VOQ^e determined by the position in which they issue (or not) a cell at time t: $VOQ^c < VOQ^e$ (also referred to as $c < e$) or $VOQ^c > VOQ^e$ (or $c > e$). $c < e$ means that at time $t-1$, VOQ^c is expected to send a cell before VOQ^e, and VOQ^c sends a cell at time t. In a similar way, $c > e$ means that at time $t-1$, VOQ^c is expected to send a cell after VOQ^e, and VOQ^c sends a cell at time t.

We also prove this lemma by contradiction. Let us take the initial time $t = 0$. By Lemma 1, input i has at least one $XP_{i,j}$ available at time N.

Initial assumption. *Assume that $VOQ_{i,j}$ is the first VOQ that has sent more than one cell within a N-slot cycle ($N \leq t < 2N$) such that it sends the second cell at time $t = 2N-1$ and the first cell was sent at a time $N \leq t \leq 2N-2$.*

$VOQ_{i,j}$ is denoted as VOQ^c. Therefore, there exists a $VOQ_{i,j'}$ (where $j' \neq j$) that is the first time that a VOQ (or the first VOQ) has not sent any cell within this N-slot period. $VOQ_{i,j'}$ is denoted as VOQ^e. Then, there are $N-2$ VOQs that have sent one cell during $N \leq t < 2N-1$. If there is a VOQ^c such that it sends a second cell at time $t = 2N-1$, there also is a VOQ^e such that either of the two cases is true: (1) $VOQ^c > VOQ^e$ or (2) $VOQ^c < VOQ^e$, where c and e are related according to Definition 7.

The two possible cases are:

1. $VOQ^c > VOQ^e$. This means that VOQ^e has $XP_{i,j'}(t) = 1$ at time $t = 2N-1$. The other VOQs have sent cells during times $N \leq t < 2N$ while $XP_{i,j'}$ has been occupied for N time slots. By Fact 2, $XP_{i,j'}$ cannot stay occupied N or more time slots; by Fact 1, $XP_{i,j'}$ should have received a cell within time $N \leq t < 2N$. This contradicts the initial assumption that $VOQ_{i,j'}$ has not sent a cell during N time slots after N time slots from the initial time.

2. $VOQ^c < VOQ^e$. This means that the crosspoint $XP_{i,j}$ related to VOQ^c has been granted (by the output arbiter) more than once within N time slots, and this can happen only if at least another input has not sent any cell competing for output j (or c), in N or more time slots. In other words, at least one crosspoint $XP_{i',j}$, where $i' \neq i$, at output j has not received a cell for N or more time slots. By Fact 4, this condition

[2] This means that the theorem is independent of the initial position of the input/output round-robin pointers, and the initial state of any $XPB_{i,j}(t)$.

[3] If a cell was sent before time $t = 0$, by Fact 2, that cell has been served by the output arbiter by time $t = N-2$, so only cells sent at time $t = 0$ or later can be considered.

contradicts the initial assumption.

Since cases (1) and (2) contradict the initial assumption, Lemma 2 is proven. Then, $VOQ_{i,j}$ issues one cell and only one within N time slots such that a different VOQ issues a cell each time slot after N time slots from the initial time.

□

Since Lemmas 1 and 2 are proved, Theorem 1 is proved.

□

Backlogged traffic at the VOQs. We can consider several initial conditions on the occupancy of VOQs. Our assumption is based on the fact that during the initial time, while CIXB-1 does not achieve 100% throughput, accumulation of cells at the VOQs can occur, independently of the initial condition. In this way, the following two cases are considered: *i)* VOQs have an occupancy larger than zero at initial time, and *ii)* VOQs have null occupancy at initial time. Since these two cases are considered, the procedure leads to the same conclusion. Therefore, by Lemma 2, the backlogged assumption traffic remains valid.