On the Maximum Throughput of a Combined Input-Crosspoint Buffered Packet Switch

Roberto Rojas-Cessa, Zhen Guo, and Nirwan Ansari

***Abstract*— Combined input-crosspoint buffered (CICB) packet switches have been of research interest in the last few years because of their high performance. These switches provide higher performance than input-buffered (IB) packet switches while requiring the crosspoint buffers run at the same speed as that of the input buffers in IB switches. Recently, it has been shown that CICB switches with one-cell crosspoint buffers, virtual output queues, and simple input and output arbitrations, provide 100% throughput under uniform traffic. However, it is of general interest to know the maximum throughput that a CICB switch, with no speedup, can provide under admissible traffic. This paper analyzes the throughput performance of a CICB switch beyond uniform traffic patterns and shows that a CICB switch with one-cell crosspoint buffers can provide 100% throughput under admissible traffic while using no speedup.**

***Index Terms*— Buffered crossbar, throughput, stability, fluid model, matrix decomposition.**

I. Introduction

The high performance of combined input-crosspoint buffered (CICB) packet switches has been of research interest for the last few years [1]-[6]. These switches provide higher switching performance than input-buffered (IB) packet switches while having the working speed of the crosspoint buffers as relaxed as that of the buffers in IB switches. CICB switches use time efficiently because input and output port selections are performed separately. That makes them attractive for implementing switches with high-speed ports.

In this paper, we follow the common practices of having incoming variable-size packets, which are segmented into fixed-size cells at the ingress side of a switch and reassembled at the egress side before leaving the switch, and of using a crossbar.

Buffered crossbars have been considered for high-performance switching in the past; however, at the expense of having large-size crosspoint buffers needed to store those packets that cannot be forwarded to the output because of contention [1]. CICB switches with first-in first out (FIFO) queues, as input queues, reduce the crosspoint-buffer size and packet loss ratio [2]. However, a CICB switch with input FIFOs may have the throughput limited by the head-of-line (HOL) blocking phenomenon. CICB switches with virtual output queues (VOQs) at the inputs can provide 100% throughput under uniform traffic using weightless [5] and weighted [4] [6] arbitration schemes. We refer to a VOQ-CICB switch as a CICB switch for the sake of brevity in the remainder of this paper.

To the best of our knowledge, CICB switches have been shown to provide 100% throughput under uniform traffic [5], [6]. However, it is of general interest to know the maximum throughput that a CICB switch, with no speedup, can provide under admissible traffic. Addressing this void, this paper demonstrates that a CICB switch with one-cell crosspoint buffers and no speedup can provide 100% throughput under admissible traffic that complies with the strong law of large numbers (SLLN) [7]. The intuition of this result is based on the knowledge that CICB switches provide higher performance than IB switches [5],[6], and that IB switches can provide 100% throughput under admissible traffic with no speedup [7], although with a high-complexity matching scheme.

To develop this proof, we use the CICB switch's property of performing input and output arbitrations separately, and analyze the following set of conditions. Input and output arbitrations can provide sufficient conditions for 100% throughput if: 1) the buffered crossbar has a crosspoint available for any input at any time slot, and 2) every input is able to send a cell of backlogged traffic to an available crosspoint at any time slot.

In addition, we show examples of input and output arbitration schemes that satisfy the above conditions, and therefore, the combination of these provide 100% throughput under any admissible traffic pattern. Of special interest is the output arbitration scheme, which uses a combination of distributed and centralized processes.

This paper is organized as follows. Section II introduces the switch and fluid models, and some preliminary definitions. Section III presents the throughput analysis of a CICB switch. Section IV presents input and output arbitration schemes that provide 100% throughput. Section V presents the conclusions.

II. Switch Model

Figure 1 shows a buffered crossbar switch with N inputs and outputs. In this switch model, there are N VOQs at each input. A VOQ at input i that stores cells for output j is denoted as $VOQ_{i,j}$. A crosspoint element in the buffered crossbar 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 $XP_{i,j}$. The buffer at $XP_{i,j}$ is denoted as $XPB_{i,j}$, and it is considered of one-cell size. Therefore, the transmission and arbitration delays are considered negligible, without loss of generality.[1] $XPB_{i,j}^{Busy}$

R. Rojas-Cessa and N. Ansari are with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, University Heights, Newark, NJ, 07102, E-mail: {rrojas, ansari}@njit.edu.

Z. Guo was with ECE Dept. at NJIT. He is now with Conexant Systems Inc., NJ, E-mail: zhen.guo@conexant.com.

[1] A larger XPB size would allow non-negligible transmission delays.

denotes an occupied XPB. The occupancy of $VOQ_{i,j}$ at up to time slot n is denoted as $Z_{i,j}(n)$. The cumulative number of packets that have arrived at $VOQ_{i,j}$ by time slot n is denoted as $A_{i,j}(n)$, and the cumulative number of packets that have departed from $VOQ_{i,j}$ by time slot n is denoted as $D_{i,j}(n)$.

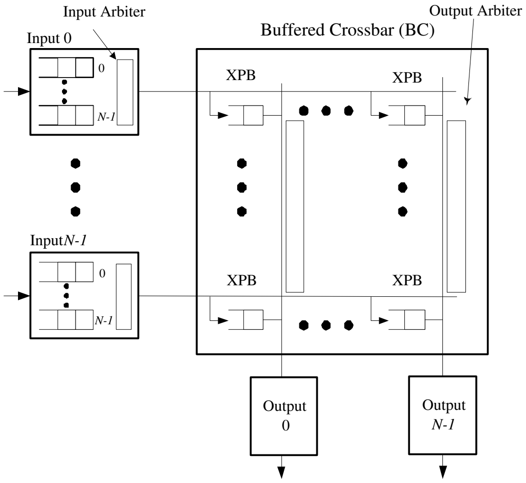

Fig. 1. Combine input-crosspoint buffered crossbar switch.

In a CICB switch, the input arbitration at input i selects a cell from a non-empty VOQ, whose corresponding XPB is available, to be forwarded to the buffered crossbar (this VOQ is said to be uninhibited). At the same time, the output arbitration selects a cell out of those XPBs with cells for output j to be dispatched from the buffered crossbar. We consider that the output arbitration can adopt either a distributed or a centralized approach as this arbitration can be implemented in the buffered-crossbar chip.

We use a fluid model [7] to analyze the properties of the VOQs in a CICB switch with no speedup and look at the stability property of this switch under a traffic model with the restrictions of being admissible and where the cell arrivals follow the strong law of large numbers: $\lim_{n\to\infty}\frac{A_{i,j}(n)}{n}=\lambda_{i,j}$, where $\lambda_{i,j}$ is the average arrival rate at $VOQ_{i,j}$.

An input arbitration uses scheme m, such that the selected VOQs can be expressed by matrix $\pi^{m}_{i,j}(n)\in\Pi$ at time slot n. For π, let $T^{m}_{\pi}(n)$ be the cumulative amount of time that a combination π has been used by time slot n as a result of arbitration. Therefore, $D_{i,j}(n)$ is the number of departures from $VOQ_{i,j}(n)$ up to time slot n, where $D_{i,j}(0)=0$ is defined.

Definition 1: If $\lim_{n\to\infty}\frac{D_{i,j}(t)}{n}=\lambda_{i,j}$, the switch is said to be rate stable. It has been proved that a switch is rate stable if the corresponding fluid model is weakly stable [7].

For $n\geq 0$, the switch dynamics are represented as:

$$Z_{i,j}(n)=Z_{i,j}(0)+A_{i,j}(n)-D_{i,j}(n) \tag{1}$$

and

$$\sum_{\pi\in\Pi}T^{m}_{\pi}(n)=n, \tag{2}$$

where $T^{m}_{\pi}(.)$ is non-decreasing.

A switch under traffic that complies with SLLN can be represented through a fluid model [7].

Definition 2: The fluid model of a switch is said to be weakly stable if for every fluid model solution (D,T,Z) with $Z(0)=0$, $Z(t)=0$ for almost every $t\geq 0$ [7].

The dynamics of the fluid model of the switch can be expressed as

$$Z_{i,j}(t)=Z_{i,j}(0)+\lambda_{i,j}t-D_{i,j}(t), \tag{3}$$

and

$$\dot{D}_{i,j}(t)=\sum_{\pi\in\Pi}\pi_{i,j}\dot{T}^{m}_{\pi}(t), if Z_{i,j}(t)>0, \tag{4}$$

where $T^{m}_{\pi}(.)$ is non-decreasing and $\sum_{\pi\in\Pi}T^{m}_{\pi}(t)=t$. Here, $\dot{g}(t)$ is the derivative of a function $g(t)$ at t.

Fact 1: (Lemma 1 in [7]) Let $f:[0,\infty)\to[0,\infty)$ be an absolutely continuous function with $f(0)=0$. Assume that $\dot{f}(t)\leq 0$ for almost every t such that $f(t)>0$ and f is differentiable at t. Then, $f(t)=0$ for almost every $t\geq 0$.

By the fluid behavior of the VOQs', for a weakly stable switch there must exist an $f(t)$, where $f(t)=0$ implies $Z(t)=0$ for every $t>0$, and $f(0)=0$ implies $Z(0)=0$.

III. Throughput Analysis of a CICB Switch

Theorem 1: A CICB, with a VOQ structure at the inputs and using no speedup, provides 100% throughput under admissible traffic.

Proof: The CICB switch is analyzed in two separated parts. The first part is concerned with the inputs, and the second part with the buffered crossbar.

As in [7], let's define

$$C_{i,j}(t)=L_{i}(t)+M_{j}(t), \tag{5}$$

where $L_i(t)=\sum_k Z_{i,k}(t)$ denotes the total amount of fluid queued at the input i at time t, and $M_j(t)=\sum_k Z_{k,j}(t)$ denotes the total amount of fluid destined for output j at time t. In other words, $C_{i,j}$ denotes the total amount of fluid at input i and the fluid destined to output j.

Since input and output arbitrations work separately in a CICB switch, if cell c is dispatched from $VOQ_{i,j}$ and is stored at $XPB_{i,j}$, then $L_i(t)$ and $M_j(t)$ decrease by one. Therefore, $C_{i,j}(t)$ is reduced by two in a single time slot.

In a similar way, as in [7], let Q be a $N\times N$ matrix with each entry being 1. Then,

$$C(t)=QZ(t)+Z(t)Q, t\geq 0 \tag{6}$$

where $C_{i,j}$ is an element of $C(t)$.

We define $f(t)$ as:

$$f(t)=\langle Z(t),C(t)\rangle=\sum_{i,j}Z_{i,j}(t)C_{i,j}(t). \tag{7}$$

where for two matrices A and B of the same size $\langle A,B\rangle=\sum_{i,j}A_{i,j}B_{i,j}$. After substituting $C_{i,j}(t)$ in (7):

$$f(t)=\sum_{i,j}Z_{i,j}\sum_{k}\left(Z_{i,k}(t)Z_{k,i}(t)\right)$$

or

$$f(t)=\sum_{i,j,k}\left(Z_{i,j}(t)Z_{i,k}(t)+Z_{i,j}(t)Z_{k,j}(t)\right). \tag{8}$$

It follows that $f(t) \geq 0$ for $t \geq 0$ and $f(0) = 0$. It is easy to see that $f(t) = 0$ implies $Z(t) = 0$. Next, we show that $f(t) > 0$ implies $\dot{f}(t) \leq 0$ for almost every t.

From (8), it follows that

$$\dot{f}(t) = 2 \sum_{i,j} Z_{i,j}(t) \dot{C}_{i,j}(t). \tag{9}$$

Therefore, $\dot{f}(t) \leq 0$ if and only if $\dot{C}_{i,j}(t) < 0$. As mentioned above,

$$\dot{C}_{i,j}(t) = \sum_k \lambda_{i,k} + \sum_k \lambda_{k,j} - 2 \tag{10}$$

where $\sum_j \lambda_{i,j} \leq 1$ and $\sum_i \lambda_{i,j} \leq 1$ make $\dot{C}_{i,j}(t) \leq 0$. Therefore, from (9) and (10), $\dot{f}(t) \leq 0$ whenever $f(t) > 0$.

The existence of $f(t)$ and the validity of Fact 1 establish that the fluid model of a CICB switch with one-cell crosspoint buffer is weakly stable as long as no input is inhibited from sending a cell to the buffered crossbar in a time slot. Then, it remains to complete the proof of Theorem 1 with the following lemmas.

In (4), the arbitration scheme m selects a VOQ such that an XPB at j receives one cell, as expressed by (5). We state the following lemma about the non-inhibition of an input arbiter:

Lemma 1: At any time slot, input i has at least an available $XPB_{i,j}$ under admissible traffic such that inhibition is avoided.

Proof:

Lemma 1 can be rephrased in terms of the output arbitration scheme, as follows:

Lemma 2: There exists an output arbitration scheme such that the selection result causes $\sum_j XPB^{Busy}_{i,j} < N$ for admissible traffic, at any time slot.

Consider the following propositions, presented in [8]. *Von Neumman proposition: if a matrix $B = (B_{i,j})$ is doubly substochastic, then there exists a doubly stochastic matrix $\bar{B}$ such that $B_{i,j} < \bar{B}_{i,j} \; \forall \; i, j$.*

Birkhoff's proposition: for a doubly stochastic matrix $\bar{B}$, there exists a set of positive numbers ϕ_k and permutation matrices P_k, where $1 \leq k \leq K$, such that $\bar{B} = \sum_k \phi_k P_k$. Let e be the column vector with all its elements being 1. As $\bar{B}$ is doubly stochastic $e = \bar{B}e = \sum_k \phi_k(Pe) = (\sum_k \phi_k)e$, making $\sum_k \phi_k = 1$.

The occupancy of the one-cell crosspoint buffers in the buffered crossbar can be represented by a matrix $XPB^{Busy} = (XPB^{Busy}_{i,j})$ such that $\sum_j XPB^{Busy}_{i,j} \leq N$ and $\sum_i XPB^{Busy}_{i,j} \leq N$. Normalizing XPB^{Busy} with respect to N, the matrix is doubly substochastic. Therefore, XPB^{Busy} can be represented as doubly stochastic $\overline{XPB}^{Busy}$, such that there exist permutation matrices that indicate which $XPB^{Busy}_{i,j}$ is served at j in a time slot. $\overline{XPB}^{Busy}$ can be obtained by following Algorithm 1 in [8].

The output arbitration scheme must select a set of XPBs as indicated by P_k where the number of occupied XPBs is at least 1. In this way, service is provided such that $\sum_{i,j} P_{i,j} > 0$ (there is at least one XPB that receives service), and thus, right after the output arbitration process is performed, the remaining occupied XPBs are at most N (and at least one) fewer than those in the previous time slot. By using the permutation matrices as the set of XPBs that are selected by the output arbitration, input i has at least one XPB available at any time slot. Furthermore, because $K \leq N^2 - 2N + 2$ [8], the smallest switch size of $N = 2$ has $K \geq 1$. Therefore, this result holds for $N \geq 2$. As the permutations correspond to a time slot, the unserved cells are held by the XPBs for the following time slot.

Since there exists an output arbitration scheme that allows inputs to be uninhibited, then Lemma 2, and consequently, Lemma 1 are proved. Therefore, Theorem 1 is proved. ■

■

Now, the matrix decomposition problem is reduced to finding the set of permutation matrices to keep all outputs active in a buffered crossbar. The finding of these matrices is translated into the use of an output arbitration scheme. As an example of such an output arbitration scheme that grants at least one XPB from input i and one XPB from output j, we introduce the scheme called Largest Column Occupancy (LCO), which is derived from the output arbitration scheme presented in [9].

IV. Largest Queue Occupancy (LQO) First and Largest Column Occupancy (LCO) First Arbitrations

The input arbiters are distributed and select a non-empty and non-inhibited (i.e., an XPB has available room for a cell) VOQ that has the largest queue occupancy (LQO).[2]

The output arbiter at j selects an XPB_i (or i) that has the largest column occupancy (LCO), in a scheme that is a combination of centralized and distributed processing, called hybrid approach. LCO uses two steps:

Step 1: Select an output $\{j \mid arg \; max_j \sum_i XPB^{Busy}_{i,j}\}$ and an input $\{i \mid arg \; max_i \sum_j XPB^{Busy}_{i,j}\}$. Set i and j as reserved and perform Step 1 with unreserved i, j pairs until no more pairs can be found (i.e., the number of unreserved inputs or outputs becomes zero, or else, when the remaining occupied XPBs belong to reserved inputs or outputs). Note that the values of XPB^{Busy}s of a reserved output j' also count in the estimation of occupied XPBs in selecting an input at output j, where $j \neq j'$. Then go to step 2.

Step 2: If there are unreserved outputs where at least one XPB is occupied, select an XPB^{Busy} arbitrarily from each unreserved output.

Figure 2 shows an example of the selection process that LCO performs in a 4 × 4 buffered crossbar. In this matrix, the rows represent the inputs and the columns represent the outputs. The selected crosspoint buffers are marked with an X. In Figure 2.a, the initial matrix has busy and idle crosspoints. Here, the column and the row with the largest number of busy crosspoint buffers are selected. Ties are broken by arbitrary selections. In Figure 2.b, the second largest column is selected. The row selection considers only those busy crosspoint buffers as shown. Figures 2.c and 2.d show the selection of the last two columns. In Figure 2.d, LCO selects $XPB_{2,0}$ by applying Step 2 as input 2 has been already selected in Step 1. Note that LCO selects output by output in a centralized manner.

[2] This is also known as longest queue first or LQF.

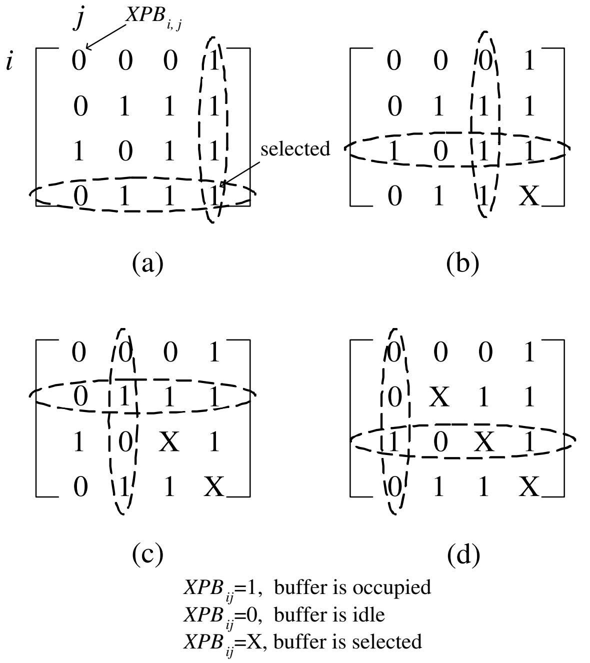

Fig. 2. Example of crosspoint selection by performing LCO in a 4×4 switch.

LCO maximizes the number of active outputs, and therefore relaxes the requirements for input arbitration schemes. Note that other input arbitration schemes can be used instead of LQO.

By computer simulation, we tested a CICB switch using LQO as input arbitration and LCO as output arbitration under traffic patterns with uniform and nonuniform distributions. The uniform traffic patterns have Bernoulli and Bursty (i.e., modulated Markov process), where the latter has average exponentially-distributed burst sizes of 10 and 100 cells. The CICB switch delivered an average cell delay with a magnitude close to that of an output buffered switch under all uniform traffic patterns. Under nouniform traffic patterns, the CICB switch delivered a finite average cell delay under unbalanced [5], diagonal[3] with $d = 0.75$, and power-of-two traffic patterns [10]. All nonuniform traffic models are admissible. All traffic models were tested with 0.99 input traffic load and in all cases 100% throughput was observed.

V. Conclusions

In this paper, we have shown that a combined input-crosspoint buffered, CICB, packet switch can provide 100% throughput with no speedup and under admissible traffic when VOQs are used at the inputs. This requires that the admissible input load follows the strong law of large numbers. The fact that input and output arbitrations in a CICB switch are performed separately allows us to analyze the queues at the inputs of the CICB switch while considering an output arbitration at the buffered crossbar that keeps inputs uninhibited (i.e., an input is able to send one cell each time slot), and to analyze the buffered crossbar while assuming that an input arbitration can select any VOQ that has a crosspoint buffer available. This paper's practical examples of input and output arbitration schemes, LQO and LCO, satisfy both conditions. LCO is a combination of distributed and centralized processes that can be implemented in the buffered crossbar chip.

Acknowledgement

This work is supported in part by National Science Foundation under Grant Awards 0435250 and 0423305.

References

[1] 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.

[2] 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.

[3] 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.

[4] M. Nabeshima, "Performance evaluation of Combined Input and Crosspoint-Queued Switch," *IEICE Trans. on Commun.*, Vol. E83-B, No. 3, March 2000.

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

[6] T. Javadi, R. Magill, and T. Hrabik, "A High-Throughput Algorithm for Buffered Crossbar Switch Fabric," in Proc. *IEEE ICC 2001*, pp. 1581-1591, June 2001.

[7] J. G. Dai and B. Prabhakar, "The Throughput of Data Switches with and without Speedup," in Proc. *IEEE INFOCOM 2000*, pp.556-564, March 2000.

[8] C-S. Chang, W-J. Chen and H-Y. Huang "Birkhoff-Von Neumann Input Buffered Crossbar Switches," in Proc. *IEEE INFOCOM 2000*, pp.1614-1623, March 2000.

[9] L. Mhamdi and M. Hamdi, "MCBF: a high-performance scheduling algorithm for buffered crossbar switches," *IEEE Commun. Letters*, Vol. 7, Issue 9, pp. 451-453, September 2003.

[10] A. Bianco, M. Franceschinis, S. Ghisolfi, A.M. Hill, E. Leonardi, F. Neri, R. Webb, "Frame-based Matching Algorithms for Input-queued Switches," in Proc. *IEEE HPSR 2002*, pp. 69-76, 2002.

[3] Represented as $\rho(i,j) = d\rho$ for $i = j$, and $(1-d)\rho$ for $j = (i+1) \bmod N$, where d is the diagonal probability and ρ is the input load.