Strictly Non-Blocking Conditions for the Central-Stage Buffered Clos-Network

Feng Wang, *Student Member, IEEE,* and Mounir Hamdi, *Senior Member, IEEE*

***Abstract*— We consider using the Clos-network to scale high performance routers, especially the *Space-Memory-Space* (SMS) packet switches. In circuit switching, the Clos-network is responsible for pure connections and the internal links are the only blocking sources. In packet switching, however, the buffers cause additional blockings. In this letter, we first propose a scalable packet switch architecture that we call the Central-stage Buffered Clos-network (CBC). Then, we analyze the memory requirements for the CBC to be strictly non-blocking, especially for emulating an output-queuing packet switch. Results show that even with the additional memory blockings the CBC still inherits advantages from the Clos-network, e.g., modular design and cost efficiency.**

***Index Terms*— Space-memory-space switching, Clos-network.**

I. Introduction

MOST current high performance routers are based on the input-queuing (IQ) or combined-input-output-queuing (CIOQ) switch architecture [1]. Years of research, however, have witnessed the difficulties in designing practical scheduling algorithms for them to provide quality-of-service (QoS). In fact, it is proven impossible to achieve this goal *without* internal speedups [2]. One key reason behind this limitation is that the buffers are *not* efficiently shared if we put the majority of them in the input side. In the IQ/CIOQ switches, each input buffer is dedicated for only one input and cannot be shared with others. On the contrary, QoS researches have been developing under the assumption of shared buffering.

Therefore, we start our work from the *Space-Memory-Space* (SMS) switch architecture, where the buffers are shared in the middle, though in a distributed way. Related research stemmed from the seminal work by Iyer [3], under another name single-stage buffered (SB) router. A generic SMS/SB switch architecture is shown in Figure 1. The switch has N inputs and N outputs, with M independent memories sandwiched in the middle. Incoming packets are *immediately* switched into one of the central memories via the left crossbar and wait for their turns to be switched out via the right one.

Compared with the IQ/CIOQ switches, packets in the SMS switch *actively* choose to buffer in the middle, whereas in the IQ/CIOQ switches, packets are *passively* buffered in the input side. This is the intuitive advantage of the SMS switch over the IQ/CIOQ switches, especially for providing QoS.

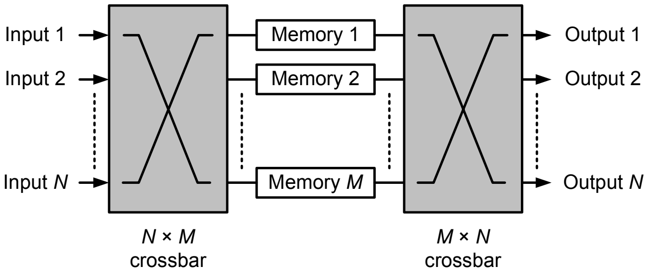

Fig. 1. The generic SMS switch architecture.

Previous researches of the SMS switch mainly focused on its scheduling algorithms, assuming the *space* to be an easy-to-configure non-blocking switching component, e.g., a crossbar. The next generation routers, however, are expected to support thousands of switching ports [4]; leading industry products include Cisco's CRS-1. As a consequence, this assumption encounters severe *scaling* problems. First, the building cost for a crossbar is on the order of $O(N^2)$, which is very expensive when N becomes large. Secondly, with current fabrication technology, the *maximum* ports a single chip can support are limited to 128×128 using FPGA, or 512×512 if using ASIC.

To scale routers for packet switching, we focus on redesigning the *space* parts in the SMS switch. In this letter, we made two main contributions as follows:

1) We first propose a Central-stage Buffered Clos-network (CBC) to make the space parts in the SMS switch scalable. We analyze the lower bound memory requirements for the CBC to be strictly non-blocking, in particular, to emulate an output-queuing switch.
2) We then present a triangularly configured CBC in practice and prove that it can use the lower bound memories to achieve strictly non-blocking for any sequence of incoming packets. We show that the CBC can significantly reduce hardware costs while providing the same packet switching capacity as that of a sing-stage crossbar.

II. The Central-stage Buffered Clos-network

The Clos-network was well studied first by Clos in the *circuit* switching area [5]. In general, it is built from multiple stages, each stage containing a variable number of small switching modules. The very property of the Clos-network is that each module pair between adjacent stages is connected by *one and only one* link.

We employ the three-stage Clos-network to construct the SMS switch. As shown in Figure 2, we duplicate the central modules (CM) in the Clos-network and link them by independent memories. We call the resulting switch architecture the Central-stage Buffered Clos-network, namely the CBC.

Manuscript received October 22, 2007. The associate editor coordinating the review of this letter and approving it for publication was X. Zhang. This work was supported in part under the Hong Kong RGC Grant HKUST6200/02E.

The authors are with the Department of Computer Science and Engineering, the Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong (e-mail: {fwang, hamdi}@cse.ust.hk).

Digital Object Identifier 10.1109/LCOMM.2008.071752.

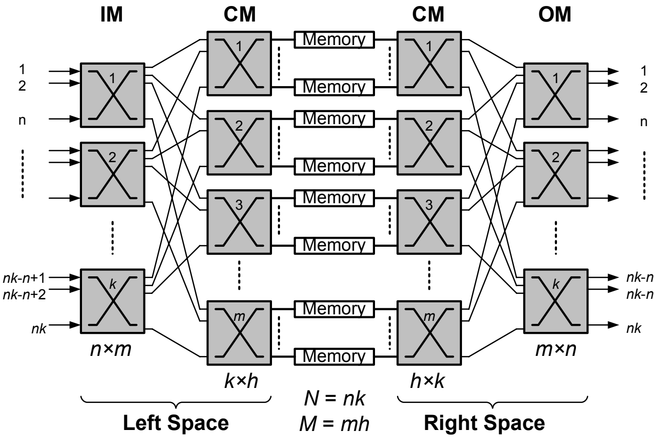

Fig. 2. The Central-stage Buffered Clos-network.

We use a quaternion (n, m, k, h) to describe the CBC. In detail, there are k input modules (IM), each of which is an $n \times m$ switch. There are two copies of m central modules (CM), each of which is a $k \times h$ (or $h \times k$ in the right space) switch. The number of independent memories linking the two copies of CMs is $M = mh$ in total. There are k output modules (OM), each of which is an $m \times n$ switch. Each pair of IM and CM (CM and OM) is connected by *one and only one* link. There are totally N inputs and N outputs for the whole CBC, where $N = nk$. Each CM copy, together with all the IM (OM), forms a *space* in the corresponding SMS switch. Every incoming packet to the CBC is immediately switched through the left space, and then stays in one of the central memories waiting for its turn to be switched out of the right space.

We adopt the fixed-length packet concept and assume all packets of the same length throughout our presentation. This is common practice in high performance routers; variable-length packets are segmented as they arrive, carried across the switch as fixed-length packets, and reassembled back into original packets before they depart.

III. Strictly Non-blocking Condition in CBC

For the Clos-network in the circuit switching, the well known strictly non-blocking condition is stated as follows:

Theorem(Clos[5]): *The three-stage Clos-network is strictly non-blocking if the number of CM* $m \geq 2n - 1$.

We note that the pure Clos-network is only responsible for establishing I/O connections/circuits and the links between adjacent stages are the only blocking sources. In the CBC, however, memories can become additional blocking sources.

To make it clear, we assume no speedup in the CBC, i.e., all components run at the outside line rate, and the memory can only read and write one packet in one time slot. In particular, we can have the following definitions.

Memory Blocking - For any memory in the CBC, if it is required to read or write more than one packet in one time slot, we say there is a memory blocking.

Link Blocking - For any link in the CBC, if it is required to transfer more than one packet in one time slot, we say there is a link blocking.

A common way to analyze a packet switch is to make it emulate an output-queuing (OQ) switch that is regarded to have optimal throughput-delay performance. By saying *emulation*, we mean that the departure time of packets from the CBC is identical with those from a shadow OQ switch fed with an identical incoming traffic. In the following analysis, we assume a packet gets its *departure time* in the shadow OQ switch before it is switched into one of the central memories.

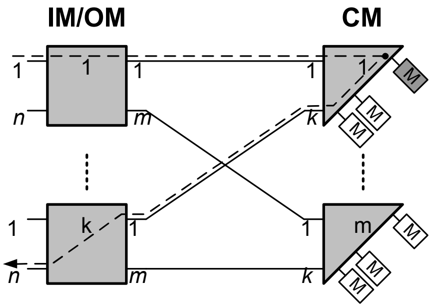

Fig. 3. The triangular CBC with $h = k$.

In this letter, we focus on analyzing the boundary conditions on the *number* of memories in the CBC. The *size* of each memory is much related to specific schedulers employed, which is out of the scope of this letter. The following theorem shows the *minimum* number of memory requirements for the CBC to remove the above two types of blockings.

Theorem 1: *The minimum number of central memories for the CBC to be strictly non-blocking, i.e., to emulate an OQ switch without internal speedup, is* $M \geq (2n - 1)(2k - 1)$.

Proof: Assume any incoming packet *P*. *P* should first select a CM to go without the link blockings. Then, *P* can select a memory in that CM without the memory blockings.

When *P* is to select a CM, it has two concerns. In the left space, other simultaneously arriving packets in the same IM as *P* may block up to $n - 1$ links, thus making up to $n - 1$ CMs unavailable to *P*. In the future right space, those packets with the same departure time and the same destined OM as *P* may also occupy up to $n - 1$ CMs. Therefore, to remove all the possible link blockings for *P*, the total number of CMs should be at least $2(n - 1) + 1 = 2n - 1$.

When *P* is to select a memory from the selected CM, it also has two concerns. The other simultaneously arriving packets from the same left CM as *P* will block up to $k - 1$ memories. In addition, the other packets to go through the same right CM as *P* in a same future time will block up to $k - 1$ memories. Therefore, to remove all the possible memory blockings, the total number of memories in each CM should be at least $2(k - 1) + 1 = 2k - 1$.

Summing them up, the total number of memories in all the CMs should be at least $(2n - 1)(2k - 1)$ to make the CBC non-blocking for any packet, i.e., emulate an OQ switch.

IV. Non-blocking Condition in Triangular CBC

The CBC is symmetric. In practice where the input ports and output ports are actually built in the same line cards, we often fold the architecture to save hardware costs. Even further, the CM in the CBC can be built in a triangular way, as shown in Figure 3. A packet flow is also shown in the Figure.

The triangular CM requires a square switch module. That is to say, when building the triangular CBC, we have this constraint: $h = k$. The following theorem states that the triangular CBC can still employ the minimum memories to remove all the blockings.

Theorem 2: *The triangular CBC can be strictly non-blocking, i.e., emulate an OQ switch without internal speedup, if the number of central memories* $M \geq (2n-1)(2k-1)$.

Proof: Refer back to Figure 2. We set $h = k$ in the CBC. Each arriving packet to be written into one of the memories faces two types of conflicts.

For the first type of conflict, we call it *arrival conflict*. It can be further divided into two kinds of conflicts due to the memory and link blockings. Consider an arbitrary packet A arriving at one of the k IMs. For other packets simultaneously arriving at the same IM as A (at most $n-1$), they can occupy at most $n-1$ CMs, where A cannot go due to the link blockings. Thus, they make up to $(n-1)h = (n-1)k$ memories unavailable to A. For packets arriving at other IMs, which add up to $n(k-1)$, they may occupy up to $n(k-1)$ memories and thus make them blocked to A. Summing them up, there are at most $(n-1)k + n(k-1)$ memories in total into which A cannot be placed due to this arrival conflict.

For the second type of conflict, we call it *departure conflict*. It can also be further divided into two kinds of conflicts. Consider the same packet A as above. For packets having the same departure time and destined OM as A (at most $n-1$), they can occupy at most $n-1$ CMs, where A cannot go due to the link blockings. Thus, they make up to $(n-1)h = (n-1)k$ memories unavailable to A. For packets having the same departure time as A but destined to other OMs, which add up to $n(k-1)$, they may occupy up to $n(k-1)$ memories and thus make them blocked to A. Summing them up, there are at most $(n-1)k + n(k-1)$ memories in total into which A cannot be placed due to this departure conflict.

Therefore, using the pigeonhole principle, the number of memories in the central stage follows:

$$\begin{aligned} M &\geq 2[(n-1)k + n(k-1)] + 1 \\ &= 4nk - 2k - 2n + 1 \\ &= (2n-1)(2k-1) \end{aligned}$$

V. Discussions

We make some discussions on advantages from the triangular CBC based on the results from above theorems.

Cost efficiency- The CBC can construct a strictly non-blocking *packet* switch with less costs than a crossbar of the same capacity. Here, we calculate the number of crosspoints in the triangular CBC by assuming each module a simple crossbar. We have seen that the least but sufficient memories in the middle is $M = (2n-1)(2k-1)$. Therefore, the least but sufficient number of the CMs is $m = M/k = (2n-1)(2k-1)/k$. It is also easy to see the number of crosspoints in each triangular CM is $k(k+1)/2$.

Summing up the crosspoints in all the modules, the total number of crosspoints in the triangular CBC is:

$$\begin{aligned} CP &= k \cdot nm + m \cdot k(k+1)/2 \\ &= kn(2n-1)(2-1/k) + (2n-1)(2-1/k)k(k+1)/2 \\ &= 4n^2k + 2nk^2 - 2n^2 - k^2 - nk - (k-1)/2 \end{aligned}$$

In normal CBC configurations, we often set $n = k$, which leads to $n = \sqrt{N}$, since $N = nk$. Therefore, the number of crosspoints becomes $6n^3 - 4n^2 - (n-1)/2 < 6n^3 = 6N\sqrt{N}$.

Since the number of crosspoints in a single-stage crossbar with the same switching capacity is N^2, it is easy to see that the triangular CBC saves significant costs in terms of the crosspoints when $N \gg 36$.

For the full CBC in Figure 2, we can set $m = 2n-1$ and $h = 2k-1$ according to theorem 1. Setting $n = k = \sqrt{N}$, the total crosspoints are less than $12N\sqrt{N}$. Compared with N^2, the full CBC saves costs when $N \gg 144$.

Smaller speedup- A common way to compensate the hardware insufficiency in practice is to employ speedup, i.e., making the internal switching components run faster than the outside line rate. Chuang [2] proved that for a CIOQ switch to emulate an OQ switch, the lowest speedup is $2 - 1/N$, which also tells that even the strictly non-blocking Clos-network ($m = 2n-1$) is used to replace the single-stage crossbar in the CIOQ switches, the lowest speedup is still $2 - 1/N$. In the triangular CBC, we have seen that the least and sufficient number of CMs to make it able to emulate an OQ switch is $m = M/k = (2n-1)(2-1/k)$. That is to say, if the triangular CBC employs the strictly non-blocking Clos-network configuration ($m = 2n-1$), the lowest speedup can be $2 - 1/k = 2 - 1/\sqrt{N}$, which is slightly *smaller* than $2 - 1/N$. Intuitively, the CBC breaks the factor N into $\sqrt{N}$ by actively buffering packets inside the switch, not passively having packets wait in the input side.

Modular design- The CBC naturally inherits the advantage of modular design from the Clos-network. That is to say, the switching components are module based and one malfunctioning will not affect others. For example, if one IM/OM is broken, other IM/OM can still keep working. And if we use redundant CMs (i.e. more than $(2n-1)(2-1/k)$), the outage of a CM will not decrease the switching capacity of the whole CBC (excluding the packets already in that CM).

VI. Conclusions

In this letter, we proposed the CBC to scale the SMS packet switches. Strictly non-blocking conditions were derived for the CBC to emulate an output-queuing switch. In addition, the CBC is module design and proven cost-efficient.

The SMS switches are believed more promising to provide QoS than the IQ/CIOQ switches, since the central memories can be made well shared. We believe that the two theorems we proved on the CBC shed some insights into building next generation scalable routers based on the SMS packet switch architecture.

References

[1] N. McKeown, "The iSLIP scheduling algorithm for input-queued switches," *IEEE/ACM Trans. Networking*, vol. 7, pp. 188-201, 1999.

[2] S. T. Chuang, A. Goel, N. McKeown, and B. Prabhakar, "Matching output queuing with a combined input output queued switch," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1030-1039, 1999.

[3] S. Iyer, R. Zhang, and N. McKeown, "Routers with a single stage of buffering," in *Proc. ACM SIGCOMM*, pp. 251-264, 2002.

[4] H. J. Chao, "Next generation routers," invited paper, *Proc. IEEE*, vol. 90, no. 9, pp. 1559-1564, Sept. 2002.

[5] C. Clos, "A study of non-blocking switching networks," *Bell Systems Techn. J.*, pp. 406-424, 1953.