Scheduling Multicast Traffic in a Combined Input Separate Output Queued Switch [1]

Ximing Hu, Xingming Zhang, Binqiang Wang, and Zhengrong Zhao

National Digital Switching System Engineering &Technological R&D Center
NO. 783 P.O.Box 1001, 450002, Zhengzhou, Henan, P.R. China
{ximinghu, zhengrong_zhao}@gmail.com

Abstract. Although several promising multicast solutions have been proposed till now; however, the support of multicasting still remains notoriously difficult for switches or routers in networks because of the traffic expansion due to multicast replication. In this paper, we propose to use a Combined Input Separate Output Queued Switch (CISOQ for short) to achieve high performance when loaded with multicast traffic. By giving novel definitions for the waiting time and the queue occupancy of multicast cells, we extend the use of oldest cell first (OCF) and longest queue first (LQF) algorithms from the unicast-only traffic load to the multicast traffic load. Furthermore, we show that 100% throughput can be obtained by a CISOQ switch when it is scheduled by OCF and LQF without speedup or by any maximal matching algorithms, just used in the unicast-only traffic load before, with a speedup of 2. The only assumptions on the multicast traffic pattern are that it is *multicast-admissible* and *SLLN* and that it does not *oversubscribe* any inputs or outputs. As far as we know, this result is the first theoretical analysis of multicast traffic arrival process till now.

1 Introduction

Nowadays, multicast-dependent services, such as multiparty telephony, video-conferencing, distributed data processing and work-group applications, are expected to share a significant portion of network applications, and ineluctably, will generate tremendous amount of multicast traffic in networks. Along with the development of next-generation network (NGN), multicasting will definitely becomes an important feature for any future switching systems designed for working in NGN.

For routers or switches in networks, the traffic expansion due to multicast replication will degrade their switching performances which are achieved in the case of unicast-only load. In order to support multicast traffic with much less or no performance degradation under heavy multicast traffic load, several promising solutions [1] have been proposed. General speaking, these proposals can be classified into four kinds according to the means of multicast duplication: the first kind is that multicast cells are replicated at input ports before they are imported into VOQ so that the indi-

[1] This work is supported by the National High-Tech Research and Development Program of China (863 Program) under grant number 2004AA103130.

vidual duplications are switched through the fabric as unicast cells [2]; the second alternative is to take advantage of the resource in switch fabrics to achieve multicast replication [3], [4]; the third kind is a hybrid replication scheme, where part of the replication occurs in the input ports and part occurs in the fabric [5]; the fourth kind is to use additional switching paths that allow the parallel transfer of multicast cells to their destinations (e.g., Multicast-enable Protocol Agnostic Forwarding Engine in [6]).

However forcible these theoretical multicast solutions may be, they have not made much difference to the way switches or routers are built due to their notoriously unpractical implementation complexities, which has been thoroughly discussed by F. M. Chiussi and A. Francini (Refer to [1] for details.) In fact, most switches and routers are still put up on the assumptions that multicast traffic constitutes a relatively small part of the total traffic, and that the distribution of multicast destinations is rather benign [1]. Intuitively, the resulting multicast performance is far from satisfaction.

In order to find a more tractable and practical switch architecture for supporting multicast traffic, we propose a new Combined Input Separate Output Queued switch (CISOQ) in this paper. The scope of our discussion is restricted to the standalone switch fabric based on bufferless crossbar. But all the results presented qualitatively apply to highly distributed switching architectures (e.g., MSM fabric arrangement in [7]). The remaining parts of this paper are organized as follows. Section 2 presents the architecture of a $N \times N$ CISOQ. Section 3 describes the graph model of the cell scheduling problem for CISOQ, and extends the OCF and LQF in [8] from the unicast-only traffic load to the multicast load. In Section 4, both simulation results and theoretic analyses are provided to evaluate the multicast performance of CISOQ when it is scheduled by OCF and LQF and maximal matching algorithms. Finally, in Section 5, we offer some concluding remarks and topics for future studies.

2 A CISOQ Switch

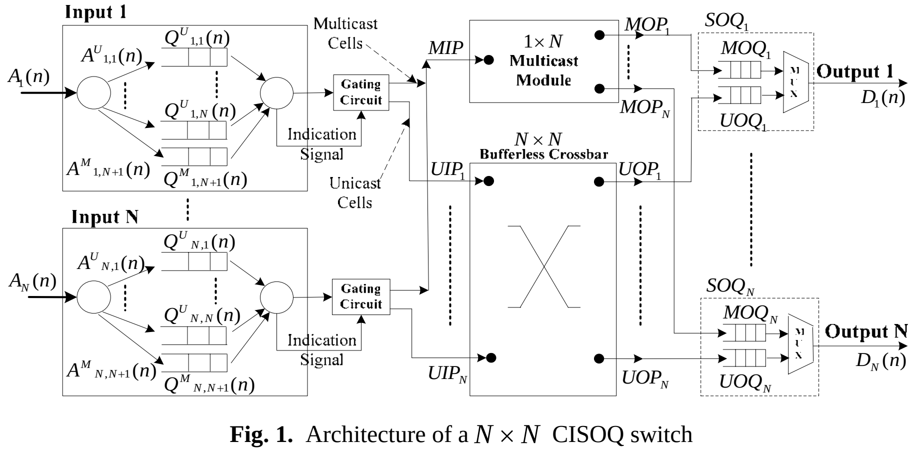

Fig. 1. Architecture of a $N \times N$ CISOQ switch

Consider the $N \times N$ CISOQ [2] in Fig. 1, connecting N inputs to N outputs. At the beginning of time slot n, either zero or one cell (unicast cell or multicast cell) arrives at input $i(1 \le i \le N)$. Each unicast cell $A^U_{i,j}(n)$ $(1 \le i \le N, 1 \le j \le N)$ contains both an unicast identifier and an unicasting-destination identifier that indicates which Unicast Output Port UOP_j $(1 \le j \le N)$ it is destined for. When $A^U_{i,j}(n)$ destined for UOP_j arrives at input i, it is immediately placed in the queue $Q^U_{i,j}$. Correspondingly, each multicast cell $A^M_{i,N+1}(n)$ $(1 \le i \le N)$ contains both a multicast identifier and a multicasting-destination identifier that indicates its set of destined Multicast Output Ports $\{MOP_j : 1 \le j \le N\}$. When $A^M_{i,N+1}(n)$ arrives at input i, it is directly placed in the queue $Q^M_{i,N+1}$ no matter what multicasting-destination identifier is. This input queuing scheme overcomes the HOL blocking of unicast cells in the same way as the VOQ scheme presented in [8]. In each time slot, each arbiter in every input $i(1 \le i \le N)$ selects no cell or one cell from the HOL of either $Q^U_{i,j}$ or $Q^M_{i,N+1}$ according to the decision made by the central scheduler. Then, an Indication Signal is produced in accord with the unicast identifier or multicast identifier of the selected cell in order to control the selected unicast cell or multicast cell to be forwarded into either the Unicast Input Port UIP_i $(1 \le i \le N)$ or the Multicast Input Port MIP.

The $N \times N$ bufferless crossbar is responsible for the proper transfer of $A^U_{i,j}(n)$ from UIP_i to UOP_j. In a parallel manner, $A^M_{i,N+1}(n)$ is duplicated and multicasted from MIP to $\{MOP_j : 1 \le j \le N\}$ by a $1 \times N$ multicast module. The output queue architecture for output j $(1 \le j \le N)$ is divided into two separate parts where $A^U_{i,j}(n)$ from UOP_j is placed into Unicast Output Queue UOQ_j and duplication of $A^M_{i,N+1}(n)$ from MOP_j is placed into Multicast Output Queue MOQ_j. The separation between UOQ_j and MOQ_j may be logical or physical. We call the resulting architecture as Separate Output Queue SOQ_j in this paper. Clearly, an output mechanism is needed in SOQ_j to regulate the access to the output line.

The approach about how to design and implement multicast modules is available and simple. Please refer to [13] for details. There is no need for us to repeat it again.

As shown in Fig. 1, by means of redundant multicast module to realize multicast replication, CISOQ brings along with it the advantages that there is neither explosion in the number of VOQ nor in the number of backpressure entities compared with the scenario where multicast replication occurs in the switch fabric. There are no needs to increase the speed of the VOQ or the speed of the switch fabric by a factor of N compared with the case where multicast replication occurs before VOQ. So, what can be concluded from a practical viewpoint is that the architecture of CISOQ is highly scalable and suitable to be deployed in the switches or routers with gigantic capacities.

[2] Without loss of generality, we assume that a CISOQ switch is fixed-size cell based. And, time is divided into time slots, equaling to the transmission time of one cell.

Furthermore, Section 4 will show that the multicast performance of CISOQ is remarkably well too, even under heavy multicast load.

3 Multicast Scheduling Algorithms for CISOQ

3.1 Graph Model for the Scheduling Problem

In each time slot, the cell scheduling problem on the $N \times N$ CISOQ can be modeled as finding maximum weight matching (Clearly, a maximum size matching is just a special case of the maximum weight matching with all edges associated with weight 1.) on bipartite graph $G = (V, E)$, where $V = V_I \cup V_O$, the set of inputs $V_I = \{i : 1 \le i \le N\}$, $V_O = \{UOP_j : 1 \le j \le N\} \cup \{MIP\}$, $|V_I| = N$ and $|V_O| = (N+1)$, E={edges between vertices of V_I and V_O }. Concretely, an edge between i and UOQ_j, associated with weight $w_{i,j}^U(n)$, represents the connection request of the unicast cell at the HOL of $Q_{i,j}^U$. An edge between i and MIP, associated with weight $w_{i,N+1}^M(n)$, represents the request of the multicast cell at the HOL of $Q_{i,N+1}^M$.

3.2 Maximum Weight Matching

In [8], two maximum weight matching algorithms: oldest cell first (OCF) and longest queue first (LQF) have been discussed just under the condition of unicast-only load. In this paper, we extend the use of OCF and LQF from the unicast-only load to the multicast load by giving novel *Definition 1* for the waiting time of the multicast cells in $Q_{i,N+1}^M$ and *Definition 2* for the queue occupancy $L_{i,N+1}^M(n)$ of $Q_{i,N+1}^M$.

Definition 1: At time slot n, the waiting time $W_{i,N+1,l}^M(n)$ of M_l in $Q_{i,N+1}^M$ $(1 \le i \le N)$ equals $\lceil \beta_l((n - s_l)m_l) \rceil$, where $M_l (0 \le l \le L)$ denotes the l^{th} multicast cell in $Q_{i,N+1}^M$, M_1 is the cell which arrives $Q_{i,N+1}^M$ just at time slot n, M_L is the cell at the HOL of $Q_{i,N+1}^M$, and let $M_0, m_0 = 0$ when there is no cell in $Q_{i,N+1}^M$, $m_l (2 \le m_l \le N)$ is the number of destined Multicast Output Ports of M_l, $s_l(1 \le s_l \le n)$ is the time when M_l arrived at input i, $\beta_l (\frac{1}{m_l} \le \beta_l \le 1)$ is a QoS coefficient for multicast traffic.

Consider the situation when there is a $N \times N$ CISOQ scheduled by OCF and M_l is queuing in $Q_{i,N+1}^M$.Suppose M_l need to be multicasted to m_l output ports which means m_l duplications of M_l need to be transmitted. If M_l has waited in $Q^{M}_{i,N+1}$ for $(n-s_l)$ slots, it has the same effect that every duplication has been waited for $(n-s_l)$ slots too. So, the total waiting time of these m_l duplications equals $((n-s_l)m_l)$. While, in the same input i, if the cell arrived at slot s_l was an unicast cell U_l instead of M_l, the waiting time of U_l would equal $(n-s_l)$. In order to prevent the waiting time of M_l increase too faster than U_l, which may leads to heavy throughput degradation of unicast traffic, $((n-s_l)m_l)$ ought to be multiplied by a coefficient $\beta_l\,(\frac{1}{m_l} \le \beta_l \le 1)$. As β_l decreases, the throughput of multicast traffic will be reduced. Specially, if $\beta_l = \frac{1}{m_l}$, M_l would just be viewed as an unicast one by OCF. Finally, in this paper, we just consider OCF algorithm for which the weight $w^{M}_{i,N+1}(n)$ and $w^{U}_{i,j}(n)$ is integer-valued, and $w^{M}_{i,N+1}(n)$ equals the waiting time $W^{M}{}_{i,N+1,L}(n)$ of M_l; in the same time, $w^{U}_{i,j}(n)$ equals the waiting time $W^{U}{}_{i,j}(n)$ of the unicast cell at the HOL of $Q^{U}_{i,j}$.

By means of the QoS coefficient, the provision of multicast cells' QoS can be controlled, which will be proven by simulation results in Section 4.

When the CISOQ is scheduled by LQF instead of OCF, a parallel definition of the queue occupancy $L^{M}_{i,N+1}(n)\ (1 \le i \le N)$ of $Q^{M}_{i,N+1}$ can be given.

Definition 2: At slot n, the queue occupancy $L^{M}_{i,N+1}(n)\ (1 \le i \le N)$ of $Q^{M}_{i,N+1}$ equals $\left\lceil \sum_{l=0}^{L}(\gamma_l m_l) \right\rceil$, where $m_l\,(2 \le m_l \le N)$ is the number of destined Multicast Output Ports of M_l, let $M_l\,(0 \le l \le L)$ denotes the l^{th} multicast cell in $Q^{M}_{i,N+1}$, M_1 is the cell which arrives $Q^{M}_{i,N+1}$ just at slot n, M_L is the cell at the HOL of $Q^{M}_{i,N+1}$, and let $M_0, m_0 = 0$ when there is no cell in $Q^{M}_{i,N+1}$, $\gamma_l\,(\frac{1}{m_l} \le \gamma_l \le 1)$ is a QoS coefficient for multicast traffic.

3.3 Maximal Matching

For practical use, a maximal matching algorithm is a better option than a maximum weight matching algorithm since it is easier to be implemented and possible to avoid unfairness. As can be concluded from the discuss of the graph mode for the scheduling problem on the CISOQ, iterative maximal size matching scheduling algorithms (e.g., PIM [9], iSLIP [10], DRR [11] etc) used in unicast-only load before can also be used to the case of multicast load by the CISOQ in order to find a maximal matching.

4 Performance Analyses of CISOQ

In this section, we adopt the conceptions in [12], and furthermore, we extend the definitions and results of [12] from the unicast-only load to the multicast case.

4.1 An *efficient* CISOQ

Considering the fluid model of the CISOQ shown in Fig. 1, We define $A_{i,N+1}^{M}(n)$ as the number of multicast cells that has arrived at $Q_{i,N+1}^{M}$ and $A_{i,j}^{U}(n)$ as the number of unicast cells that has arrived at $Q_{i,j}^{U}$ up to time slot n. We assume the multicast and unicast arrival processes $\{A_{i,N+1}^{M}(\cdot), A_{i,j}^{U}(\cdot), i, j = 1,\cdots,N\}$ satisfy a strong law of large numbers (*SLLN*), as proposed in [12]: with probability one,

$$\lim_{n\to\infty}\frac{A_{i,N+1}^{M}(n)}{n} = \lambda_{i,N+1}^{M}, i = 1,\cdots,N, \lim_{n\to\infty}\frac{A_{i,j}^{U}(n)}{n} = \lambda_{i,j}^{U}, i, j = 1,\cdots,N. \quad (1)$$

Where $\lambda_{i,N+1}^{M}$ is called the multicast arrival rate at $Q_{i,N+1}^{M}$ and $\lambda_{i,j}^{U}$ is called the unicast arrival rate at $Q_{i,j}^{U}$. In the following definition, we extend the notion that no inputs or outputs are oversubscribed from unicast-only load to the case of multicast traffic.

Definition 3: When loaded with multicast traffic, no inputs or outputs are said to be *oversubscribed* if

$$\forall i, \lambda_{i,N+1}^{M} + \sum_{j=1}^{N}\lambda_{i,j}^{U} \le 1, i = 1,\cdots,N, \ \forall j, \sum_{i=1}^{N}\lambda_{i,j}^{U} \le 1, j = 1,\cdots,N. \quad (2)$$

Definition 4: The multicast traffic is said to be *multicast-admissible*, if

$$\sum_{i=1}^{N}\lambda_{i,N+1}^{M} \le 1. \quad (3)$$

Definition 5: When loaded with multicast traffic, a switch operating under a matching algorithm is said to be *rate stable* if, with probability one,

$$\lim_{n\to\infty}\frac{D_{i,N+1}^{M}(n)}{n} = \lambda_{i,N+1}^{M}, i = 1,\cdots,N, \lim_{n\to\infty}\frac{D_{i,j}^{U}(n)}{n} = \lambda_{i,j}^{U}, i, j = 1,\cdots,N. \quad (4)$$

where $D_{i,N+1}^{M}(n)$ is the number of multicast cells departed from $Q_{i,N+1}^{M}$ and $D_{i,j}^{U}(n)$ is the number of multicast cells departed from $Q_{i,j}^{U}$ up to time slot n.

Definition 6: When loaded with multicast traffic, a switch is said to be *efficient* if there at least exist one scheduling algorithm which can make this switch *rate stable* for any arrival processes satisfying (1), (2) and (3).

Theorem 1: When loaded with multicast traffic, CISOQ is *efficient* when it is scheduled by LQF or OCF, as long as the speedup $s \ge 1$.

Theorem 2: When loaded with multicast traffic, CISOQ is *efficient* when it is scheduled by any maximal weight matching algorithm, as long as the speedup $s \geq 2$.

Proof: Let $\lambda(n)$ be the rate matrix of the input traffic at time slot n:

$$\lambda(n) = [\lambda^{U}_{i,j}(n) \quad \lambda^{M}_{i,N+1}(n)], i, j = 1, \cdots, N. \tag{5}$$

From *Def. 3,4*, we know λ is a class of doubly sub-stochastic non-square $N \times (N+1)$ matrices. While in the unicast-only case, λ just change into square $N \times N$ ones as in [12], and,

$$\lambda(n) = [\lambda^{U}_{i,j}(n)],\ i, j = 1, \cdots, N. \tag{6}$$

So, the proofs of *Theorem 1,2* of [12], which are carried out in the unicast-only case, can be used to prove *Theorem 1,2* of this paper as long as the definitions of both fluid model and notations used in [12] are improved from square $N \times N$ matrices to non-square $N \times (N+1)$ ones by the same way as λ above. We omit the details here.

An *efficient* CISOQ can keep each output link 100% busy. From the long-run fraction of time viewpoint, an *efficient* CISOQ can achieve 100% throughput, if it has infinite buffer capacities. However, for practical use, a CISOQ with finite buffer capacities can approach *efficient* by means of an elaborate queuing discipline which is closely tied with CISOQ architecture and will be the topic of our forthcoming paper.

4.2 Simulation Result

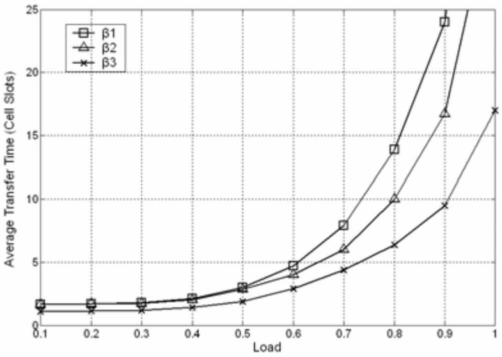

Fig. 2. Simulation result of a 32×32 CISOQ

The simulation is carried out in a 32×32 CISOQ scheduled by OCF, which is loaded with both multicast and unicast traffic. The multicast destinations of multicast cells are uniformly distributed, but the total number of destined Multicast Output Ports of every multicast cell is fixed. Let $\lceil \beta_3 m_3 \rceil > \lceil \beta_2 m_2 \rceil > \lceil \beta_1 m_1 \rceil = 1$. While with the ratio of multicast load increasing from 0.1 to 1, the performance of CISOQ turns worse more slowly as the value of $\lceil \beta \cdot m \rceil$ increases, as shown in Fig. 2.

5 Conclusion

In this paper, we have presented the Combined Input Separate Output Queued (CISOQ) switch, which is a new switch fabric configuration achieving 100% throughout as long as the multicast traffic load satisfies the assumption that it is *multicast-admissible* and *SLLN* and that there is no inputs or outputs to be *oversubscribed*. And, we are not aware of any analytical studies of multicast traffic arrival processes prior to this work. Obviously, the assumption applies to very general multicast traffic, so the results intrinsically have high practical significance.

Indeed, the provision of QoS guarantees for unicast and multicast traffic mainly depends on two key factors: one is the scheduling algorithm that arbitrates the transfer of cells prepared at the HOL of each input port across the switch fabric; the other is the queuing discipline that is responsible to prepare cells according to certain requirements of QoS within each VOQ and resolve the conflicts occurring among HOLs of all VOQs in each input port. Till now, in this paper, we have just concentrated on the scheduling algorithm pertaining to CISOQ. The queuing discipline which is closely tied with CISOQ will be the topic of our forthcoming paper.

References

1. F. M. Chiussi and A. Francini, “Scalable Electronic Packet Switches,” IEEE J. Select. Areas Commun., Vol. 21, No. 4, (May 2003) 486–500
2. Chen X., Lambadaris I., Hayes J., “A general unified model for performance analysis of multicast switching,” in Proc. IEEE GLOBECOM’92, New York, Vol. 3, (1992) 1498–502
3. W. Chen, Y. Chang, “A high performance cell scheduling algorithm in broadband multicast switching systems,” in Proc. IEEE GLOBECOM’97, New York, Vol. 1, (1997) 170–4
4. B. Prabhakar, N. McKeown, R. Ahuja, “Multicast scheduling for input queued switches,” IEEE J. Select. Areas in Commun., Vol. 15, No. 5, (June 1997) 855–66
5. M. Ajmone Marsan, F. M. Chiussi, A. Francini, et al, “Compression of multicast labels in large input-queued IP routers,” IEEE J. Select. Areas Commun. Vol. 21, (2003) 21-30
6. M. Song, J. Song and H. LI, “Improved Multicast Traffic Scheduling Scheme in the Packet-Switching Systems,” Journal of China Universities of Posts and Telecommunications, Vol. 11, (Sep. 2004) 1–7
7. F. M. Chiussi and A. Francini, “A Distributed Scheduling Architecture for Scalable Packet Switches,” IEEE J. Select. Areas Commun., Vol. 18, No. 12, (Dec. 2000) 2665–2683
8. N. Mckeown, A. Mekkittikul, V.Anantharam and J. Walrand, “Achieving 100% Throughput in an Input-Queued Switch,” IEEE Trans. Commun., Vol.47, (Aug. 1999) 1260–1267
9. T. Anderson, S. Owicki, J. Saxie, and C. Thacker, “High speed switch scheduling for local area networks”, ACM Trans. Comput. Syst., Vol. 11, No. 4, (Nov. 1993) 319–352
10. N. McKeown, “The iSLIP scheduling algorithm for input-queued switches”, IEEE/ACM Trans. on Networking, Vol. 7, No. 2, (April 1999) 188–201
11. J. Chao, “Saturn: a terabit packet switch using dual round-robin”, IEEE Communication. Magazine .December, (2000) 78–84
12. J. Dai and B. Prabhakar, “The throughput of data switches with and without speedup”, in Proc. of IEEE INFOCOM’2000, (May 2000) 556–564
13. Gua, Ming-Huang and Ruay-Shiung Chang, “Multicast ATM switches: survey and performance evaluation,” Computer Communication Review, Vol. 28, No 2, (1998) 98–131