

# Stability of CIOQ Switches with Finite Buffers and Non-Negligible Round-Trip Time

M. Gusat, F. Abel, F. Gramsamer, R. Luijten, C. Minkenberg, and M. Verhappen  
IBM Research, Zurich Research Laboratory, 8803 Rüschlikon Switzerland  
gusat@ieee.org

**Abstract**—We propose a systematic method to determine the lower bound for internal buffering of practical CIOQ switching systems. To this end we introduce a deterministic traffic scenario that stresses the global stability of finite output queues. We demonstrate its usefulness by dimensioning the buffer capacity of the CIOQ under such traffic patterns. Compliance with this property maximizes the performance achievable with finite buffers.

**Keywords:** switching, backpressure, admissible, stability, RTT

## I. INTRODUCTION

Most recently proposed switching fabrics belong to the class of combined input-output queued (CIOQ) architectures [4-17]. As shown in Fig. 1, CIOQs involve a degree of internal speedup and output queue buffering. The two main classes of CIOQs are centralized with limited speedup [9-17] and distributed with full speedup [4-8]. While the speedup required for ideal output-queue (OQ) emulation [10] has been the subject of numerous studies, e.g., [2,11-17], the topic of buffering has received little attention so far. Whereas the majority of theoretical papers assume infinite buffers, practical CIOQ implementations have a limited amount of internal buffering. Independently of their physical implementation, OQ buffers can be logically managed as shared (SM) [4-8] or as partitioned/dedicated memory architectures, e.g. CICQ [23,24].

Investigations of CIOQ buffering requirements, such as [3], are less numerous, and assume negligible round-trip times (RTT). Our contribution is that we investigate the switch core behavior under increasingly large RTTs.

A question worth asking is: Can one derive a lower bound for the buffering capacity, and can one find a benchmark and a metric for assessing the global queuing capacity required by a CIOQ system? As we will show in the following, the answer is positive if we augment the definition of work-conservation with global stability.

**Definition (Work-Conservation Property):** A work-conserving (WC) switch will serve any output for which at least one packet is present in the system.

However, the WC definition is too strict to be practical in assessing the properties of CIOQ switches with finite buffers; formally; no such system can be strictly WC [1]. The consequences deriving from this result are that neither is perfect OQ emulation feasible nor is an absolutely robust and traffic-agnostic CIOQ switch physically possible. If strict work conservation is unachievable, then it follows that the notion of

strict work conservation has no practical value in sizing the buffers. Therefore we propose the notion of *absolute global stability* (AGS) of a CIOQ core; this sets a tight upper limit to the global queue buildup. Based on this property we derive the lower bound for the internal buffer requirements. To derive a pragmatical instrument for buffer dimensioning, we propose a new benchmark scenario, called sweeping hotspot. From the outset we make the assumption that a switching core must be lossless under *any* traffic pattern, and its throughput ( $T_{\text{put}}$ ) should emulate as closely as possible that of an ideal OQ switch. In [21] is shown that traffic patterns and behavior for the Internet are not predictable, motivating the assumption that the switch fabric should be traffic independent.

The remainder of the paper is organized as follows. In Section II we present the background and set the framework to this paper. In Section III we prove that a lower bound for the buffering capacity of CIOQ switching systems exists. To this end we will first outline our method and introduce the sweeping hotspot traffic scenario. We then prove that the absolute global stability theorem places a tight bound on the number of packets admitted to the switch without violating work-conservation. As a result, we prove that the output buffer size for both SM and CICQ must scale as  $O(\text{RTT} \cdot N^2)$ . The degree of AGS is proposed as the quantitative metric to differentiate between various CIOQ implementations. We conclude with future work.

## II. BACKGROUND AND FRAMEWORK

### A. Background

In addition to the study of the internal buffering capacity performed in [3], we credit [1,2,22] as predeceasing investigations in the same direction as our current work, namely, analysis of stability and work-conservation. However, a number of basic assumptions distinguish our approach from previous studies. First, from [22] we borrow and extend the notions of admissible and inadmissible traffic [15,22]. However, as the traffic pattern proposed here is deterministic, basic calculus provides us with exact results. This contrasts with [22], where the use of ergodic traffic called for the use of stochastic methods. Second, whereas for the investigation of PPS [2] assumes infinite OQ buffers and that “no state information is communicated from the core to the inputs,” we apply a different set of hypotheses for the global stability study of a single-plane CIOQ. As any practical switch contains a finite OQ capacity, OQ state information must be communicated periodically to the IQ schedulers in order to prevent OQ over-

flow. In CIOQs with centralized arbitration, such state information is implicitly included in the scheduling/arbitration algorithm [9-12]. In CIOQs with distributed scheduling, the state information can be conveyed, e.g., as backpressure (BP). A more sophisticated flow-control scheme prevents both overflow and underflow.

While [1] assumes a CIOQ with finite OQs, for the purpose of that study “backpressure is applied instantaneously in case an OQ is completely full”. In fact, many existing CIOQs assume fractional RTTs [5], or disregard the RTT altogether. However, we argue for the opposite trend. In current CIOQ systems, the distance between IQs and OQs spans tens to hundreds of feet, whereas the cycle time has shrunk from microseconds per packet to a few nanoseconds. One can no longer neglect the transport latencies, which affect both the datapath and the control path. Furthermore, scheduling performance, scalability in number of chips and power budget per system favor the support of RTT inside the CIOQ core. In our study the sum of all logical and physical delays are lumped into the RTT. BP is characterized by the fundamental time constant of a closed-loop feedback control system,  $\tau = \text{RTT}$ , which marks the delay between the *issuance* of a BP command and its *effect* becoming visible at the same location where it was issued. We assume arbitrarily large RTT values *within* the switching fabric, normalized to packet cycles, in the range of tens to hundreds.

### B. Framework

A typical CIOQ system is shown in Fig. 1. The CIOQ system under study contains an  $N \times N$  switch core organized as single stage and single plane. We start by developing the global stability conditions of a CIOQ with full output speedup  $S_o = N$ ; then we generalize the results to CIOQs with any speedup  $S$ . We seek to derive stability conditions without constraining the CIOQ speedup or scheduling. The system contains a total of  $N^2$  VOQ input queues [9-17] and  $N$  OQs; its physical and logical OQ architecture could be implemented as shared-memory [4,8], distributed [23,24] or a combination thereof [5,6]. Time-slotted operation with fixed sized packets is assumed.

An implication of the results of [1] asserts that “more buffer space at the output is always better”; the authors in [5] advocate for 2 to 4 times more buffering capacity than their actual implementation provides. The main benefit of a buffered switch resides in its capability to remove the *immediate dependency* between packet arrivals and departures; provided there is sufficient internal queuing capacity, the downstream departure processes can be arbitrarily decoupled from preceding and current upstream arrival processes. As a con-

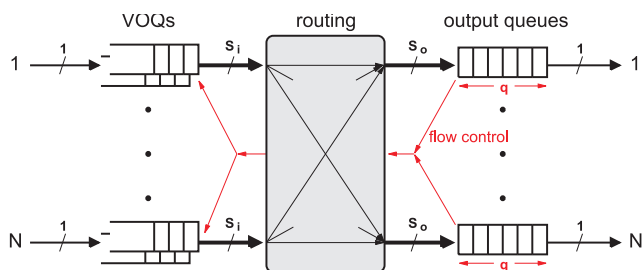


Figure 1. CIOQ system with finite buffers.

sequence, flow control, scheduling, and, essentially, the QoS service levels can be tuned to a range of throughputs and delay metrics. In short, internal buffering provides scheduling freedom, at the high expense of fast memory, i.e., low density and high power.

By definition, a WC switch must be inherently lossless. For better clarity without significantly reducing generality, we initially assume a shared-memory CIOQ architecture with On/Off backpressure. Here the goal of BP is to achieve lossless operation.

A difference between our work and previous studies is that [2] focuses on the absolute stability, which at the limit is equivalent to the work-conservation property of any *single* output (Definition 1, condition 3). We extend the method to a global CIOQ switching core across *all* its output ports. Intuitively defined, a globally stable CIOQ with finite buffers must achieve the same aggregate throughput as an ideal OQ switch with infinite buffers, if both are offered the same traffic patterns. To this end we must derive an absolute bound for the global queue buildup during a relevant, if possible, deterministic traffic scenario. The question is how to build such a scenario. This will be shown below.

### III. GLOBAL STABILITY: METHOD AND TRAFFIC SCENARIO

In this section we analyze under which conditions a CIOQ with finite buffers and distributed scheduling can emulate the ideal OQ switch with infinite buffers, when the traffic pattern exhibits correlated spatial burstiness. Our focus is on both the individual and the aggregate output behavior. In a globally stable CIOQ the arrival processes to any  $O_j$  must be independent of those of other outputs. Decoupling can be achieved by partitioning the buffers into dedicated output queues, as in a class of CICQs, whereas in shared-memory CIOQs [4-6], it is achievable by providing sufficient speedup  $S$  and capacity for *all* the outputs. We prove here that either case elicits a global internal capacity  $O(\text{RTT} * S^2)$ . More specifically, we determine the global stability conditions that would permit neither overflow nor starvation of any output. Global stability under admissible traffic can be defined as the condition that, when the aggregate drain rate of all the output queues equals that of an unconstrained ideal OQ switch, the global queuing capacity available within a CIOQ core will accommodate exactly the maximum queue buildup.

#### A. Method

Let  $\lambda_{ij}(t)$  be the instantaneous intensity of the packet-arrival process from input  $I_i$  to output queue  $O_j$ . Let  $\Lambda(t)$  be the global matrix of intensities of arrival processes, as

$$\Lambda(t) = \|\lambda_{ij}\| \quad (1)$$

With internal output speedup  $S_o = N$ <sup>1</sup>, the sum per column in (1) is the total rate of arrival processes for  $O_j$ ,  $\lambda_j(t) = \sum_i \lambda_{ij}(t)$ . However, in a buffered CIOQ we are interested in the total amount of work arriving at output  $O_j$  within a specific time interval

<sup>1</sup>Speed up is generalized in [25] to values other than  $N$ .

$$a_j(t_{\text{start}}, t_{\text{end}}) = \sum_{i=1}^N \int_{t_{\text{start}}}^{t_{\text{end}}} \lambda_{ij}(t) dt. \quad (2)$$

Next, let  $\underline{\mu}(t)$  be the instantaneous intensity of the departure process from the output queue  $O_j$ ;  $\boldsymbol{\mu}(t)$  is the vector of departure rates

$$\boldsymbol{\mu}(t) = \|\mu_{j(t)}\|, \quad j \in [1, N]. \quad (3)$$

The output service rate  $\mu_j(t)$  can be independently constrained, i.e., the service rate of an output  $O_j$  can be reduced or blocked by a condition external to our system (e.g., a blocking condition downstream). In a lossless CIOQ, if such an output is temporarily blocked while the arrival processes  $\lambda_j(t)$  are still active, no packets should be dropped; instead, backpressure mode is activated. Accordingly, a constraint applied to  $\boldsymbol{\mu}(t)$  will eventually backpressure the arrival processes in a feedback loop. Thus,  $\boldsymbol{\mu}(t)$  is considered an independent variable that influences  $\Lambda(t)$ . In general, the aggregate rate of arrival processes for any output  $O_j$ ,

$$\lambda_j(t) = \sum_i \lambda_{ij}(t) \in [0, N \cdot \sum \lambda_{\max, i}]. \quad (4)$$

Under ideal traffic, the total rate of arrival processes for any output  $O_j$  is bounded

$$0 \leq \lambda_j(t) \leq 1, \quad (5)$$

while under admissible traffic, the sum of any column of the global matrix of intensities of arrival processes,  $\lambda_j(t) < 1$ ,  $(\forall \nabla)t$ . We assume a bimodal distribution for the arrival processes; thus, either  $\lambda_{ij}(t) = \lambda_{\max} = 1$ , or  $\lambda_{ij}(t) = \lambda_{\min} = 0$ . If  $\lambda_j(t) = 0$ , e.g., because no traffic is available in the IQs, then  $O_j$  is idling by necessity. If, at the other extreme,  $\lambda_j(t) > \mu_j(t)$  for a given time interval, then  $O_j$  becomes congested and its OQ will backlog, which eventually will cause backpressure. This will occur despite a departure service with maximum rate  $\mu_{\max} = 1$ .

**Definition** We denote as *inadmissible traffic* any situation when the aggregate arrival intensity exceeds the available aggregate departure service rate,  $\lambda_j(t) > \mu_j(t) = \mu_{\max}$ .

During inadmissible traffic, even the ideal OQ switch experiences loss of throughput and queue buildup; this will be discussed later. On the other hand, whereas the ideal OQ switch has maximum throughput under admissible traffic, a real CIOQ may still experience loss of throughput and non-work-conserving behavior. One cause is that, if not dimensioned according to global stability, admissible traffic may induce an OQ to overflow, or, starve. This could, for example, result from premature backpressuring of inputs, which otherwise could provide the OQs with new arrivals. Thus, not all the traffic offered will materialize in throughput.

### B. Traffic Scenario

For the study of global stability we employ a deterministic form of admissible traffic, denoted as *sweeping hotspot*. As the name suggests, this is a deterministic traffic pattern whereby all input traffic sources synchronously target one output after another. If the  $k$ -th output is currently hotspotted, then

$$\sum_{i=1}^N \lambda_{ij}(t) = \begin{cases} N \cdot \lambda_{\max}, & j = k, \\ 0, & j \neq k. \end{cases} \quad (6)$$

Persistent contention of two, or more, arrival processes at a single  $O_j$  will eventually result in congestion and backlog the corresponding OQ. We define the *congestion rate* at the single hotspot  $O_j$  as

$$\lambda_{\text{cong}, j}(t) \triangleq \lambda_j(t) - \mu_j(t) = \sum_i \lambda_{ij}(t) - \mu_j(t). \quad (7)$$

Assume that  $O_j$  is just being hotspotted, starting at time  $t_j$ . We define the *congestion epoch*.

$$t_{\text{CE}} = \varepsilon + \tau, \quad (8)$$

where  $\varepsilon$  is the BP *activation* delay, an initial reaction time until backpressure mode is activated, and, as mentioned,  $\tau$  is the round-trip time (RTT) after which the effects of backpressure activation are felt. Correspondingly, under maximum degree  $N:1$  of hotspot congestion starting at  $t_0 = 0$ , the corresponding OQ <sub>$j$</sub>  will receive a monotonically increasing amount of work,

$$a_{\text{cong}, j}(0, t_{\text{CE}}) = \sum_{i=1}^N \int_0^{t_{\text{CE}}} \lambda_{\text{cong}, j}(t) dt, \quad (9)$$

which, if the traffic pattern does not cease, will also cause the hotspotting IQs upstream to backlog once the BP is activated. The congestion graph rooted on the congested output  $O_j$ , and buildup of OQ <sub>$j$</sub>  concatenated with its respective ingress VOQs—all fully backlogged, is denoted *saturation tree* [20].

The sweeping hotspot will periodically (minor cycle) shift the hotspot target from one output to another one; and then repeat the sequence in the next *major cycle*. A major cycle of the sweeping sequence is shown in Fig. 2.

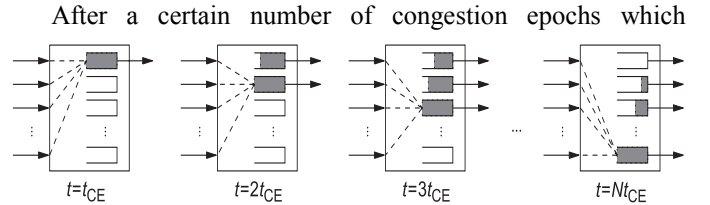


Figure 2. Graphical description of sweeping hotspot.

depends on the CIOQ speedup, such an admissible traffic pattern converges to a steady state. This particular property of the proposed benchmark for global stability enables us to test the convergence (absolute bound) of the queue buildup during back-to-back congestion periods. We will show that a CIOQ with finite buffers can emulate the ideal behavior and achieve the same steady state.

### C. Absolute Global Stability

Here we prove that during admissible congestion, the global queue buildup of a CIOQ core with finite buffers is strictly upper-bounded. Moreover, no OQ underflows, i.e., it is work-conserving, provided that the global queuing capacity available in the CIOQ core equals this absolute bound.

**Global Stability Theorem.** *If  $Q$  is the total internal buffer capacity of an  $N \times N$  CIOQ with effective RTT =  $\tau + \varepsilon$ , and,*

$Q_{\max, \text{open}}$  is the maximum of the global queue buildup with open outputs, then the system is globally stable if and only if

$$Q \geq Q_{\max, \text{open}} .$$

In particular, for a shared-memory CIOQ with  $S = N$ ,

$$Q_{\text{SM\_max, open}} = N(N-1)(\lambda_{\max} - \mu_{\max}/2)(\varepsilon + \tau). \quad (10)$$

Due to space limitations the proof is omitted here; however, it is provided in [25]. By replacing the BP activation delay  $\varepsilon$ , we derive the lower bound of global stability as:

$$Q \geq Q_{\text{SM\_max, open}} = \tau N^2/2 \quad (11)$$

□ (A)GS Theorem

The strong form (equality) of the global stability condition above denotes the *absolute* global stability. That is, the end of the drain epoch for the first (oldest) output hotspotted, i.e.,  $OQ_1$ , coincides with the termination of the  $N^{\text{th}}$  congestion epoch and with the start of new major sweep cycle, or

$$T_{\text{drain\_AGS}} = N t_{\text{CE}}, \quad (12)$$

which prevents the underflow of the oldest queue, i.e., the loss of the work-conservation property. Absolute global stability under sweeping hotspot traffic ensures that after an output was activated, it will never unnecessarily idle. Also, the global capacity according to AGS theorem is sufficient to prevent that an output can be starved due to other congested outputs; this is equivalent to non-hogging.

#### D. Discussion

In order to compare our results with [3], we will reverse an initial assumption, i.e., the one that all outputs are always fully open. This was required to establish the absolute lower bound of queue buildup during admissible congestion; in fact,  $\mu_j(t) = \mu_{\max} \forall j, t$ , is optimistic over arbitrary time intervals. In a “closed” scenario, any output  $O_j$  may be arbitrarily constrained to a service rate  $\mu_j(t) \in [0,1]$ . A conservative variation of the sweeping hotspot scenario assumes that  $\mu_j(t) = 0$  just after<sup>2</sup> the onset of a congestion epoch. In this case the local maximum of  $Q(t)$  is reached later, i.e., after  $N$  congestion epochs (after which outputs must open), during which no departures occurred:

$$Q_{\max, \text{closed}} = N^2 \lambda_{\max} (\tau + \varepsilon'), \quad (13)$$

where  $\varepsilon'$  is the BP activation delay with closed outputs, calculated similarly to the case with open outputs. However, in this case the on/off threshold is reached sooner after  $\varepsilon'$ , and, if  $\lambda_{\max} = 1$ , then

$$Q_{\max, \text{closed}} = N(N+1) \tau. \quad (14)$$

The difference between our study and the worst case from [3, Section 3.2] is that we do not assume that the congested outputs must be externally blocked. By comparison, the absolute global stability shows that a shared-memory CIOQ with finite

buffers can emulate the ideal OQ with infinite buffers with less than half of the buffer capacity resulting from [3]. Indeed, if all other factors are equal,

$$N^2/2 < N(N+1). \quad (15)$$

Next, unlike [2], where  $O_j$  is investigated independently of the other outputs, we study the stability across the full set of  $N$  outputs, by considering the dependencies arising from practical constraints and/or resource sharing.

Also it must be observed that the value  $Q_{\text{SM\_max, open}} = \tau N^2/2$ , holds only for shared-memory architectures. If partitioned per output queue, an absolute globally stable CICQ with backpressure and full speedup requires *more* capacity. In general it can be shown that the following inequalities hold

$$Q_{\text{SM\_max, open}} < Q_{\text{CICQ\_max, open}} < Q_{\max, \text{closed}}. \quad (16)$$

Because arrivals in a shared-memory CIOQ can readily use the queuing capacity just freed by the departures of other outputs, this architecture seems appealing. However, the relative benefit of shared-memory CIOQ vs. CICQ (ca. 50% less internal capacity for SM) is rather theoretical, as the shared-memory requires that the entire OQ capacity is fully-spended RAM—instead of memory operating at line speed, which is sufficient for CICQ.

Finally, we observe that, whereas in (14) we derive  $Q_{\max, \text{closed}}$  for the first  $N$  congestion epochs, i.e., the first major cycle of the sweeping hotspot, the result is of limited practical value. First, under such circumstances and with closed outputs, the ideal OQ remains *non-blocking* owing to its infinite buffers. Meanwhile its throughput is null, because the system functions as a degenerated buffered switch, i.e., as a memory. This makes the “closed output” assumption questionable. As there is no reason to stop the sweeping hotspot after the first major cycle, if it continues past the  $N$ -th congestion epoch, then the global queue buildup is unbounded in an ideal OQ switch. Unlike during the “open output” scenario, the  $Q(t)$  function is *not convergent* as long as at least one departure process is constrained; the global backlog continues to increase monotonically—or, at the best, interleaved with periods of stagnation.

If convergence to steady state is not achievable with constrained departures, we argue that a “closed” traffic scenario can neither be used as benchmark nor the global stability property as a metric; instead, this case ought to be treated as inadmissible traffic. A pragmatical target is to determine the inflexion point beyond which the performance gains are cancelled by the cost of over-provisioning with additional capacity.

#### E. Global Stability Degree

As *degree of global stability* of a CIOQ core, we introduce the normalized ratio of the global stability equations (10, 11) to the available internal OQ capacity of that CIOQ. This is a helpful metric to assess to which degree a CIOQ will not experience throughput loss, i.e., to which degree it is non-blocking under an admissible traffic pattern such as the sweeping hotspot.

For example, consider a  $64 \times 64$  shared-memory CIOQ core with full output speedup  $S = N$ , packet size and memory width

<sup>2</sup>In fact this timing is not essential; outputs could have been closed *ab initio*.

of 64 B, RTT = 50 packet cycles and port service rates  $\lambda_{\max} = \mu_{\max} = 64$  Gbps. To achieve a global stability degree of 1.0, if implemented in shared-memory, 6.25 MB of RAM with cycle time 0.06 ns are needed. Clearly, the shared-memory requires a large number of interleaved memory banks, each 512-bit wide. Despite the apparent advantage of requiring less memory capacity than the partitioned architectures, e.g, CIOQs, the implementation cost of a globally stable SM CIOQ architecture becomes prohibitive for line rates beyond 10Gbps.

#### IV. CONCLUSIONS

We have derived the exact lower bound of global stability under admissible traffic, and proved that in such conditions a practical CIOQ can emulate the ideal OQ switch. For the particular case of shared-memory CIOQs, stability is achievable with buffers of less than half the size as estimated by the previous dimensioning attempts. Moreover, we have proposed the *global stability degree* as a metric to assess the potential loss of throughput under challenging traffic patterns.

As method we have used the sweeping hotspot as a deterministic traffic benchmark to measure the stability degree. The sweeping hotspot is an important benchmark because of its unique combination of contrasting properties: locally and periodically it produces maximally contending arrival patterns of inadmissible traffic, while globally, when applied to an ideal OQ with infinite buffers, it converges to admissible traffic with steady state whereby throughput is maximal. The method proves practical in sizing any CIOQ systems that must support non-negligible RTTs within the switching core.

#### V. LIMITATIONS AND FUTURE WORK

Whereas the absolute lower bound of global stability is useful for global dimensioning of CIOQ cores, the result per se does not provide a comprehensive insight about how to partition and schedule this buffering capacity. Notably missing are more specific work-conservation issues<sup>3</sup>, i.e., does Equation (10) hold for any other traffic scenario?

Once the issue of stability has been dealt with, some more pragmatical issues arise next. (i) Is a globally stable CIOQ core feasible, and up to which size? (ii) Can we emulate its correctness properties with less expensive constructs? (iii) Can other, tighter, benchmarks be found?

#### REFERENCES

- [1] C. Minkenberg, "Work-conservingness of CIOQ packet switches with limited output buffers," IEEE Commun. Lett. (in press, 2002).
- [2] D. Khotimsky and S. Krishnan, "Stability analysis of a PPS with bufferless input demultiplexors," in Proc. ICC'2001.
- [3] M. Katevenis, "Buffer requirements of credit-based flow control when a minimum draining rate is Guaranteed," in Proc. HPCS '97, 4th IEEE Workshop on Architecture & Implementation of High Performance Communication Subsystems, Chalkidiki, Greece, June 1997.
- [4] C. Minkenberg and T. Engbersen, "A combined input- and output-queued packet-switch system based on PRIZMA switch-on-a-chip technology," IEEE Commun. Mag., vol. 38, no. 12, pp. 70-77, Dec. 2000.
- [5] M. Katevenis, D. Serpanos, and E. Spyridakis, "Switching fabrics with internal backpressure using the ATLAS I single-chip ATM switch," in Proc. GLOBECOM '97, Phoenix, AZ, Nov. 1997.
- [6] F. M. Chiussi and A. Francini, "A distributed scheduling architecture for scalable packet switches," IEEE J. Sel. Areas Commun., vol. 18, no. 12, pp. 2665-2683, Dec. 2000.
- [7] R. Fan, H. Suzuki, K. Yamada, and N. Matsuura, "Expandable ATOM switch architecture (XATOM) for ATM LANs," in Proc. ICC '94, vol. 1, New Orleans, LA, May 1994, pp. 402-409.
- [8] K. Yun, "A terabit multi-service switch with quality of service support," in Proc. HOTI8, A Symposium on High Performance Interconnects, Aug. 2000, Stanford, CA.
- [9] N. McKeown, B. Prabhakar, and M. Zhu, "Matching output queueing with combined input and output queueing," in Proc. 35<sup>th</sup> Annual Allerton Conf. Communication, Control and Computing, Monticello, Oct. 1997.
- [10] I. Stoica and H. Zhang, "Exact emulation of an output queueing switch by a combined input output queueing switch," in Proc. 6<sup>th</sup> IEEE/IFIP IWQoS '98, Napa Valley, CA, May 1998, pp. 218-224.
- [11] A. Charny, P. Krishna, N. Patel, and R. J. Simcoe, "Algorithms for providing bandwidth and delay guarantees in input-buffered crossbar switches with speedup," in Proc. 6<sup>th</sup> IEEE/IFIP IWQoS '98, Napa Valley CA, May 1998, pp. 225-234.
- [12] P. Krishna, N. S. Patel, A. Charny, and R. J. Simcoe, "On the speedup required for work-conserving crossbar switches," IEEE J. Sel. Areas Commun., vol. 17, no. 6, pp. 1057-1066, 1999.
- [13] B. Prabhakar and N. McKeown, "On the speedup required for combined input and output queued switching," Automatica, vol. 35, 1999.
- [14] S.-T. Chuang, A. Goel, N. McKeown, and B. Prabhakar, "Matching output queueing with a combined input output queued switch," IEEE J. Sel. Areas Commun., vol. 17, no. 6, pp. 1030-1039, Jun. 1999.
- [15] J. G. Dai and B. Prabhakar, "The throughput of data switches with and without speedup," in Proc. INFOCOM 2000, Tel Aviv, Israel, Mar. 2000, vol. 2, pp. 556-564.
- [16] Y. Oie, M. Murata, K. Kubota, and H. Miyahara, "Effect of speedup in nonblocking packet switch," in Proc. ICC '89, Jun. 1989, pp. 410-414.
- [17] A. K. Gupta and N. D. Georganas, "Analysis of a packet switch with input and output buffers and speed constraints," in Proc. IEEE INFOCOM '91, Bal Harbour, FL, Apr. 1991, pp. 694-700.
- [18] I. Iliadis and W. E. Denzel, "Analysis of packet switches with input and output queueing," IEEE Trans. Commun., vol. 41, no. 5, pp. 731-740, May 1993.
- [19] A. Pattavina and G. Bruzzi, "Analysis of input and output queueing for nonblocking ATM switches," IEEE/ACM Trans. Networking, vol. 1, no.3, pp. 314-328, Jun. 1993.
- [20] G. F. Pfister and V. A. Norton, "Hot spot contention and combining in multistage interconnection networks," IEEE Trans. Computers, vol. C-34, no. 10, pp. 933-938, Oct. 1985.
- [21] S. Floyd and V. Paxson, "Difficulties in simulating the Internet," IEEE/ACM Trans. Networking, vol. 9, no.4, pp. 392-403, Aug. 2001.
- [22] N. McKeown, V. T. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," in Proc. IEEE Infocom '96, vol. 1, pp. 296-302, San Francisco, Mar.1996.
- [23] T. Javidi, R. Magill, and T. Hrabik, "A high-throughput scheduling algorithm for a buffered crossbar switch fabric," in Proc. ICC 2001, Helsinki, SF, June 2001, vol. 5, pp. 1586-1591.
- [24] R. Rojas-Cessa, E. Oki, and H. Jonathan Chao, "CIXOB-k combined input-crosspoint-output buffered packet switch," in Proc. GLOBECOM '01, vol. 4, pp. 2654-2660.
- [25] M. Gusat, F. Abel, F. Gramsamer, R. Luijten, C. Minkenberg, and M. Verhappen, "A Stability Study of CIOQs with finite buffers and non-negligible RTT," IBM Research Report, RZ 3440 (2002).

<sup>3</sup> Topics such as QoS and multicast are considered orthogonal to our study.