

# On the Throughput Achievable by Isolated and Interconnected Input-Queueing Switches under Multiclass Traffic

E. Leonardi, M. Mellia, M. Ajmone Marsan, F. Neri  
Dipartimento di Elettronica, Politecnico di Torino, Italy  
e-mail: {leonardi,mellia,ajmone,neri}@mail.tlc.polito.it

*Abstract*—Many recent studies provide an extended investigation of the maximum throughput achievable in Input-Queueing (IQ) or Combined-Input-and-Output-Queueing (CIOQ) packet switches. Some scheduling policies, among which maximum weight matching algorithms, were identified as optimal, in the sense that they were proved to achieve 100% throughput under any admissible single-class traffic pattern. Most of the results in the literature, however, consider just one switch in isolation, operating on packets belonging to a single traffic class. In this paper we first generalize known results, showing that a wide class of IQ schedulers operating on multiple traffic classes can achieve 100% throughput. In addition, we address the problem of the maximum throughput achievable in a network of interconnected IQ switches loaded by multiclass traffic, and we devise some simple scheduling policies that guarantee 100% throughput when switches are interconnected in a network. Both the Lyapunov function methodology and the fluid models approach are used to obtain our results.

## I. INTRODUCTION AND PREVIOUS WORK

Great attention has been recently devoted by the research community to the design of Input Queueing (IQ) switch architectures and the assessment of their performance. IQ switching architectures have become an attractive architectural solution for the design of large-size and high-capacity switches when Anderson [1] and McKeown [2] showed in their pioneering works that the negative effects of Head-of-the-Line (HoL) blocking on performance can be reduced or completely eliminated by adopting per-destination queueing (also called virtual output queueing) at input cards.

A major issue in the design of IQ switches is that the access to the switching fabric must be controlled by some form of scheduling algorithm<sup>1</sup>, which operates on a (possibly partial) knowledge of the state of input queues. This means that control information must be exchanged among line cards, either through an additional data path or through the switching fabric itself, and that intelligence must be devoted to the scheduling algorithm, either at a centralized scheduler, or at line cards in a distributed manner.

We refer in this paper to the case of fixed-size data units, called “cells” from the ATM jargon, possibly obtained by segmenting variable-size packets (for example IP datagrams), and to a synchronous switch operation, according to which input/output connections are changed synchronously at every cell

<sup>1</sup>This work was supported by Lucent Technologies-Bell Labs under contract 575/2000 – “Performance analysis of scheduling algorithms for input-buffered packed switches”.

<sup>1</sup>The term “scheduling algorithm” for packet switching architectures is used in the literature for two different types of schedulers: switching matrix schedulers and flow-level schedulers [3], [4]. *Switching matrix schedulers* decide which input port is enabled to transmit in a non purely output-queueing switch; they avoid blocking and solve contentions within the switching fabric. *Flow-level schedulers* decide which cell flows must be served in accordance to QoS requirements. In this paper the term scheduling algorithm is only used to refer to the former class of algorithms.

time (called “slot”) for all ports.

The problem faced by scheduling algorithms with Virtual Output Queues (VOQs) can be formalized as a maximum size or maximum weight matching on the bipartite graph in which nodes represent input and output ports, and edges represent cells to be switched. Edges may be associated with weights related to the state of input queues.

In order to achieve good scalability in terms of switch size and port data rate, it is essential to reduce the computational complexity of the scheduling algorithm. This objective has been often pursued by introducing a moderate speed-up with respect to the data rate of input/output lines [5] in the switching fabric, as well as in the input and output memories. In this case, buffering is required at outputs as well as inputs, and the term “combined input/output queueing” (CIOQ) is used. Obviously, when the speed-up is such that the internal switch bandwidth equals the sum of the data rates on input lines, input buffers become useless.

Even the introduction of a moderate speed-up can have a significant cost for very high capacity switches. Thus, one of the major challenges for the design of very high capacity switching architectures is the design of low complexity scheduling algorithms that minimize the speed-up required to guarantee good performance in terms of both throughput and delay to the different information flows.

Along with the search for low complexity, highly scalable, well performing, switch architectures and scheduling algorithms, a relevant effort has been recently devoted to the identification and development of a methodology to assess the performance achievable by IQ switch architectures. A complete set of general theoretical results could indeed provide an important framework to drive applied researchers toward better performing solutions.

Two methodologies were applied to obtain most of the known theoretical results on IQ and CIOQ switches: the Lyapunov function methodology, and the fluid models methodology. The Lyapunov function methodology was applied in [6], [7], [8], to find the stability region of several scheduling algorithms under general traffic patterns. The fluid models methodology [9], [10] was used in [11] for the same purpose.

Pure IQ switches (i.e., switches with no speed-up), whose scheduling policy implements a Maximum Weight Matching (MWM) in each slot, were proved to achieve the same performance in terms of throughput of Output Queueing (OQ) switches in [6], [11], and [8], under a wide class of traffic patterns, when considered in isolation, and dealing with a single class of traffic. This result holds provided that edge weights are

proportional to the length of the corresponding VOQ (LQF policy), or to the age of the head-of-the-line cell (OCF policy) in the corresponding VOQ, or, finally, to the sum of all cells stored in the corresponding input and output ports (LPF policy) [2]. To the best of our knowledge, instead, no general result exists on the performance of pure IQ switches dealing with *multiple traffic classes*; only heuristic scheduling algorithms supporting multiple traffic classes were proposed in the recent literature [12], [13], [14], [15], and their performance was assessed by simulation for a limited number of traffic patterns.

A wider set of results are known for CIOQ switch architectures. CIOQ switch with speed-up equal to 2 have been proved to be able to exactly emulate OQ switches implementing any monotonic work-conserving queueing discipline [5]. This result holds under general traffic conditions also for switches interconnected to other switches; its practical relevance, however, is strongly limited by the very large complexity required to implement the scheduling policy. In [16] a simpler scheduling algorithm (called LOOFA), operating on a single class of packets, has been shown to ensure work conservation with speed-up equal to 2. A wide class of low complexity scheduling policies, among which maximal size matching algorithms, have been proved to achieve the same performance of OQ switches in terms of throughput in [11] and [7] with speed-up equal to 2.

The problem of guaranteeing QoS for real time traffic in CIOQ switching architectures has been the objective of recent studies [16], [17], [18], relying on the implementation of Weighted Fair Queueing schemes at inputs and outputs, and their integration with the scheduling algorithm. These works, however, contain only few and quite loose results on the performance of the proposed schemes: a speed-up equal to 3 is required for the scheme proposed in [16] to guarantee delay performance comparable to OQ under admissible leaky-bucket compliant traffic, while the scheme in [17] requires a speed-up equal to 2 to guarantee bounded delays to admissible leaky-bucket compliant traffic.

Finally, in [19] it was shown that a specific network of IQ switches implementing a MWM scheduling policy can exhibit an instable behavior when none of the switches are overloaded. This new, counterintuitive result, opened new perspectives in the research on IQ and CIOQ switches, reducing the value of most of the results obtained for switches in isolation. In [19] the authors propose a policy named LIN, that, if implemented in each switch of the network, leads to 100% throughput under any admissible traffic pattern when each traffic flow in the network is leaky-bucket compliant. The LIN policy, however, is based on a pre-scheduling of cell transmissions at each switch of the network, thus relying on an exact knowledge of the traffic pattern at each switch, and leading to large computational complexity when the traffic load approaches 1. In addition, the result proved in [19] cannot be easily extended to more general traffic patterns in which flows are not leaky-bucket compliant.

In this paper we perform a theoretical investigation of the performance achievable by switch architectures dealing with multiple traffic classes. We also focus on the performance achievable by a network of IQ switches. Our results are obtained by applying both the Lyapunov function and the fluid models methodologies. The interested reader can refer to [20] for a presentation of

the basic theoretical results that form the background necessary to our analysis.

We first show that the extension of schedulers for IQ switches to multiclass traffic leads to surprising results. For example, we show that no IQ scheduler can achieve 100% throughput in a two traffic classes environment, if strict priority is given to cells of one class with respect to cells of the other class. We then define a large class of scheduling policies that allow a pure IQ switch to achieve 100% throughput under multiclass traffic.

We then analyze the performance of a network of interconnected IQ switches, trying to provide a better understanding of the instability phenomena first presented in [19], which can occur in networks of IQ or CIOQ switches, even when each switch implements efficient scheduling policies.

The long-term objective of this study is the design of scheduling policies that guarantee good performance also when switches are interconnected in a network serving multiple traffic classes. In general, the implementation of optimal scheduling policies designed for a network of switches is rather complex, and requires a coordination among different switches, as already pointed out in [19]. However, we show that the deployment of quite a simple policy that requires a minimum amount of information to be exchanged only among neighboring nodes guarantees 100% throughput in a network of pure IQ switches.

Simple simulation results are provided to support our analytical findings.

## II. PRELIMINARY DEFINITIONS AND NOTATIONS

### A. Queueing Systems

Consider a system of  $J$  discrete-time queues (of infinite capacity) represented by row vector  $Q$ , whose  $j$ -th component,  $0 \leq j < J$ , is a descriptor associated with the  $j$ -th queue in the system. The system of queues handles  $N \geq J$  classes of customers. Each customer arrives to the network from outside, receives service at a number of queues, and leaves the network. Customers change class every time they move through the network. We suppose that each class  $k$  of customers,  $0 \leq k < N$ , univocally identifies a queue in the system at which all class  $k$  customers are enqueued, i.e., all customers of class  $k$  are enqueued at the same queue. Let  $L(k) = j$  the system location function that associates each class  $k$  of customers with the queue  $j$  at which class  $k$  customers are enqueued.  $L^{-1}(j)$  is the counter-image of  $j$  through function  $L(k)$ . In general  $L^{-1}(j)$  returns a set of customer classes. When  $N = J$ , each customer class is in one-to-one correspondence with a queue.

Let  $X_n = (x_n^{(0)}, x_n^{(1)}, \dots, x_n^{(N-1)})$  be the row vector whose  $k$ -th component  $x_n^{(k)}$ ,  $0 \leq k < N$ , represents the number of customers of class  $k$  in the system at time  $n$ . We say that the set of customers of the same class forms a virtual queue in the system of queues; thus in the paper we indicate the set of customers of class  $k$  with the term “virtual queue  $k$ ”. We suppose that the service times required by customers of all classes are deterministic and equal to one unit of time. We consider only non-preemptive atomic service policies, i.e., service policies that serve customers in an atomic fashion, never interrupting the service of the customer that is currently in service.

The evolution of the number of queued customers is described

by  $x_{n+1}^{(k)} = x_n^{(k)} + e_n^{(k)} - d_n^{(k)}$ , where  $e_n^{(k)}$  represents the number of class  $k$  customers that entered virtual queue  $k$  (and thus physical queue  $L(k)$ ) in time interval  $(n, n + 1]$ , and  $d_n^{(k)}$  represents the number of customers departed from virtual queue  $k$  in time interval  $(n, n + 1]$ .  $E_n = (e_n^{(0)}, e_n^{(1)}, \dots, e_n^{(N-1)})$  is the vector of entrances in the virtual queues, and  $D_n = (d_n^{(0)}, d_n^{(1)}, \dots, d_n^{(N-1)})$  is the vector of departures from the virtual queues. With this notation, the system evolution equation can be written as

$$X_{n+1} = X_n + E_n - D_n \quad (1)$$

The entrance vector is sum of two terms: vector  $A_n = (a_n^{(0)}, a_n^{(1)}, \dots, a_n^{(N-1)})$  representing the customers arrived at the system from outside, and vector  $T_n = (t_n^{(0)}, t_n^{(1)}, \dots, t_n^{(N-1)})$  of recirculating customers;  $t_n^{(k)}$  is the number customers departed from some virtual queue and entered into virtual queue  $k$  in time interval  $(n, n + 1]$ . Note that when customers do not traverse more than one queue (as it is typically the case for a switch in isolation), vector  $T_n$  is null for all  $n$ , and  $A_n = E_n$ .

The  $N \times N$  matrix  $R_n = [r_n^{(k,l)}]$  is the *routing matrix*, whose element  $r_n^{(k,l)}$  represents the fraction of customers departing from virtual queue  $k$  in time interval  $(n, n + 1]$  that enter virtual queue  $l$ .

We assume that the system of queues forms an *open network*, i.e.,<sup>2</sup>

$$\Gamma = I + E[R_n] + E[R_n]^2 + E[R_n]^3 + \dots = (I - E[R_n])^{-1}$$

exists and is finite, i.e.,  $I - E[R_n]$  is invertible for all  $n$ . We further assume that the routing matrix is time invariant, i.e.,  $E[R_n] = R$  does not depend on the time instant. We also impose that  $R$  satisfies the strong law of large numbers:

$$\lim_{n \rightarrow \infty} \frac{\sum_{i=0}^{n-1} R_i}{n} = R \quad \text{with probability 1}$$

Note that  $T_n = D_n R_n$ . The law of evolution of virtual queues can thus be rewritten as:

$$X_{n+1} = X_n + A_n - D_n(I - R_n) \quad (2)$$

Let us consider the external arrivals process  $A_n = (a_n^{(0)}, a_n^{(1)}, \dots, a_n^{(N-1)})$ ; we suppose that arrival processes are stationary, i.e.,  $E[A_n] = \Lambda = (\lambda^{(0)}, \lambda^{(1)}, \dots, \lambda^{(N-1)})$  does not depend on the time interval  $[n, n + 1)$ . Moreover, we suppose that arrival processes at each virtual queue satisfy the strong law of large numbers, i.e.:

$$\lim_{n \rightarrow \infty} \frac{\sum_{i=0}^{n-1} A_i}{n} = \Lambda \quad \text{with probability 1}$$

The average workload  $W$  provided at each virtual queue by customers that entered the system of queues in time interval  $[n, n + 1)$  is given on average by  $W = \Lambda(I - R)^{-1}$ .

<sup>2</sup>  $E[X]$  denotes the expectation of random quantity  $X$ .

## B. Norms and Other Operators

Before proceeding, we define two norm functions that will be helpful in the sequel.<sup>3</sup>

*Definition 1:* Given a vector  $Z \in \mathbb{R}^N$ ,  $Z = (z^{(k)}, 0 \leq k < N)$ , we call  $\|Z\|_2$  the Euclidean Norm of  $Z$ , i.e.,  $\|Z\|_2 = \sqrt{\sum_{k=0}^{N-1} (z_n^{(k)})^2}$ .

*Definition 2:* Given a vector  $Z \in \mathbb{R}^N$ ,  $Z = (z^{(k)}, 0 \leq k < N)$ , and a location function  $L(k) = j$ , from  $0 \leq k < N$  to  $0 \leq j < J$ , with  $J \leq N$ , norm  $\|Z\|_{\max L}$  is defined as

$$\|Z\|_{\max L} = \max_{j=0, \dots, J-1} \left\{ \sum_{k \in L^{-1}(j)} |z^{(k)}| \right\} \quad (3)$$

The name stands for maximum queue length.

To simplify our notation, we define a matrix associated with queue length vectors, which will be often used in the remainder of the paper.

*Definition 3:* Given vector  $X \in \mathbb{R}^N$ , the  $N \times N$  diagonal matrix  $\mathcal{I}[X]$  is such that  $\mathcal{I}^{(j,j)}[X]$  is equal to 1 if the  $j$ -th component of  $X$ ,  $x^{(j)}$ , is non-null, and it is null otherwise.  $\mathcal{I}^{(i,j)}[X] = 0$  when  $i \neq j$ .

## C. Stability Definitions for a System of Queues

Several definitions of stability for a network of queues can be found in the technical literature. We recall here some of them.

*Definition 4:* A system of queues achieves *100% throughput* if

$$\lim_{n \rightarrow \infty} \frac{X_n}{n} = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} (E_i - D_i) = 0 \quad \text{with probability 1}$$

where  $X_n$  is the queue lengths vector at time  $n$ . A system that achieves 100% throughput is also said *rate stable*.

*Definition 5:* A system of queues is *weakly stable* if, for every  $\epsilon > 0$ , there exists  $B > 0$  such that

$$\lim_{n \rightarrow \infty} P\{\|X_n\| > B\} < \epsilon$$

where  $P\{E\}$  denotes the probability of event  $E$ .

*Definition 6:* A system of queues is *strongly stable* if

$$\lim_{n \rightarrow \infty} \sup E[\|X_n\|] < \infty$$

Any norm can be used in the two definitions above.

Note that strong stability implies weak stability, and that weak stability implies 100% throughput. Indeed, the 100% throughput property allows queue lengths to indefinitely grow with sub-linear rate, while the weak stability property entails that the servers in the system of queues are able to process the whole offered load, but the delay experienced by customers can be unbounded. Strong stability implies, in addition, the boundedness of average customer delays.

<sup>3</sup> In this paper,  $\mathbb{N}$  denotes the set of non negative integers,  $\mathbb{R}$  denotes the set of real numbers, and  $\mathbb{R}^+$  denotes the set of non negative real numbers.

A necessary condition for the system of queues to achieve stability is that the average workload provided at each queue by customers entering the system of queues in time interval  $[n, n + 1)$  does not reach 1. This condition, that we call *no-overload condition*, is also a sufficient condition for stability in any BCMP type network of queues [21]. This condition can be formalized as:

$$\|W\|_{\max L} < 1$$

In general, as shown in [19], this condition does not guarantee the stability of a generic network of queues.

### III. ONE SWITCH IN ISOLATION WITH MULTICLASS TRAFFIC

#### A. Notation

We consider IQ or CIOQ cell-based Switches with  $P$  input ports and  $P$  output ports, all running at the same cell rate (and we call them  $P \times P$  IQS or CIOQS). The switching fabric is assumed to be non-blocking and memoryless, i.e., cells are only stored at switch inputs and outputs.

At each input, cells are stored according to a Multi-Class Virtual Output Queuing (MCVOQ) policy: one separate queue is maintained at each input for each output and for each traffic class. We suppose that cells belonging to  $C$  different traffic classes arrive at input (and output) ports. Thus, the total number of input queues in each switch is  $N = CP^2$ . We do not model possible output queues since they never become unstable under admissible traffic patterns.

With respect to the definitions of Section II, we must keep a difference between *traffic* classes, and *customer* classes in the network of queues: we map cells belonging to a given traffic class onto different customer classes which depend on the VOQ at which cells are enqueued. According to the definitions of Section II, we have a single traffic class when  $J = N$  (the number of VOQs equals the number of customer classes).

The switch in isolation can be modelled as a system comprising  $N$  virtual queues. Let  $q^{(k)}$ ,  $k = CPi + Cj + l$  be the virtual queue at input  $i$  storing cells of class  $l$  directed to output  $j$ , with  $i, j = 0, 1, 2, \dots, P - 1$  and  $l = 0, 1, 2, \dots, C - 1$ .

We define three functions referring to VOQ  $q^{(k)}$ :

- $I(k)$ : returns the index of the input card in which the VOQ is located
- $O(k)$ : returns the index of the output card to which VOQ cells are directed
- $C(k)$ : returns the index of the traffic class associated with the VOQ.

We consider a synchronous operation, in which the switch configuration can be changed at slot boundaries. We call *internal time slot* the time necessary to transmit a cell from an input toward an output. We call instead *external time slot* the duration of a cell on input and output lines. The difference between external and internal time slots is due to the switch speed-up, and to possibly different cell formats (e.g., due to additional internal header fields).

At each internal time slot, the switch scheduler selects cells to be transferred from input queues to output queues. The set of cells to be transferred during an internal time slot must satisfy two constraints: i) at most one cell can be extracted from the

MCVOQ structure at each input, and ii) at most one cell can be transferred toward each output, thus resulting in a correlation among servers activities at different queues.

We adapt the definition of  $\|Z\|_{\max L}$  to the case of the single switch handling multiclass traffic as follows.

*Definition 7:* Given a vector  $Z \in \mathbb{R}^N$ ,  $Z = (z^{(k)})$ ,  $k = CPi + Cj + l$ ,  $i, j = 0, 1, \dots, P - 1$ ,  $l = 0, 1, \dots, C - 1$ , the norm  $\|Z\|_{IO}$  is defined as:

$$\|Z\|_{IO} = \max_{j=0, \dots, P-1} \left\{ \sum_{k \in I^{-1}(j)} |z^{(k)}|, \sum_{k \in O^{-1}(j)} |z^{(k)}| \right\}$$

The constraint on the set of cells transferred through the switch can be formalized in the following manner.

*Definition 8:* At each time slot, the scheduler of an IQS selects for transfer from queues  $Q = (q^{(k)})$  a set of cells denoted by vector  $D \in \mathbb{N}^N$ ,  $D = (d^{(k)}) \in \{0, 1\}$ ,  $k = CPi + Cj + l$ ,  $i, j = 0, 1, \dots, P - 1$ ,  $l = 0, 1, \dots, C - 1$  so that  $\|D\|_{IO} \leq 1$ . Set  $D$  is said to be a set of non-contending cells, or a switching vector.

In order not to overload any input and output switch port, the total average arrival rates in cells/(external slot) must be less than 1 for all input and output ports; in this case we say that the traffic pattern is *admissible*.

*Definition 9:* The traffic pattern loading an (isolated) IQS is admissible if and only if  $\|E\|_{IO} = \|\Lambda\|_{IO} < 1$ , where  $E$  is the stationary average of  $E_n$ .

Note that any admissible traffic pattern can be transferred without losses in an output buffered switch architecture with infinite queues.

#### B. Main Results for a Switch in Isolation

In [6] and [11], it has been proved, using two different approaches that IQ switches subject to a single traffic class can achieve 100% throughput under a wide class of arrival processes.

In this section we extend the discussion to IQ switches operating on multiple traffic classes. We first show that the extension of schedulers for IQ switches to the multiclass case leads to the surprising result that no IQ scheduler can achieve 100% throughput with two traffic classes when strict priority is given to cells of one class. We then define a wide class of scheduling policies that allow the switch to achieve 100% throughput in a multiclass environment.

We say that a two-class IQ scheduler gives strict priority to class A cells with respect to class B cells if the presence of class B cells in the switch VOQs does not cause any perturbation to the transfer of class A cells.

Let us consider the traffic pattern described in Fig. 1, in which flows  $0 \rightarrow 0$  and  $2 \rightarrow 2$  have higher priority with respect to flows  $1 \rightarrow 0$  and  $1 \rightarrow 2$ . Suppose that the cell arrival process associated with each input/output flow (we have 4 flows in the example) is Bernoulli: a cell arrives in each slot with probability  $p$ . Note that for every  $p < 1/2$ , the traffic pattern loading the switch is admissible. Since contention can never arise among high priority cells stored in input queues, high priority cells are transferred toward output ports with no delay. Thus, whenever two high priority cells arrive at inputs 0 and 2 in the same slot

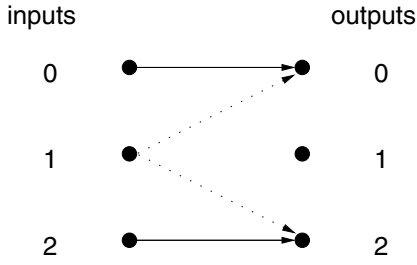


Fig. 1. Scenario in which any IQS implementing a strict priority discipline cannot achieve 100% throughput

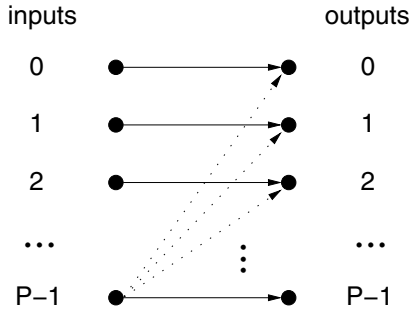


Fig. 2. Scenario in which any CIOQ switch with speed-up smaller than  $2 - 1/P$  implementing a strict priority discipline cannot achieve 100% throughput

$n$ , they are immediately transferred toward switch outputs, and no low priority cell can be transferred. One low priority cell can be transferred toward its output only when one or no high priority cell arrives. Since two high priority traffic cells arrive in the same slot with probability  $p^2$ , the maximum throughput achievable by lower priority cells is  $1 - p^2$ . Thus, whenever  $2p > 1 - p^2$  (i.e.,  $p > \sqrt{2} - 1$ ), the switch does not achieve 100% throughput.

We now generalize the previous considerations to a multiclass environment. Let us consider a multiclass CIOQ switch operating according to a strict priority discipline.

*Theorem 1:*  $2 - 1/P$  is the minimum speed-up  $S$  required to achieve 100% throughput in a  $P \times P$  CIOQ switch handling multiclass cells according to a strict priority rule.

*Proof: Necessity.* Let us consider the traffic pattern described in Fig. 2, in which flows  $i \rightarrow i$ , with  $0 \leq i < P$ , have higher priority with respect to flows  $P - 1 \rightarrow i$ , with  $0 \leq i < P - 1$ . Suppose that the high priority arrival process at input  $i$ , with  $0 \leq i < P - 1$ , is Bernoulli, with probability  $p = (P - 1)/P$ . Let us further suppose that high priority cells arrivals at input  $i$ ,  $0 \leq i < P - 1$ , are correlated in such way that in each slot either no high priority cells arrive at the switch, or  $P - 1$  high priority cells arrive at the switch, one at each input  $i$ , with  $0 \leq i < P - 1$ . Finally, high priority cells arrive at input  $P - 1$  with rate  $q = 1/P - \epsilon$ , but they can arrive only when no other higher priority cells arrive at other inputs. Low priority cell arrivals are described by independent Bernoulli processes, with probability  $q = 1/P - \epsilon$ . It is immediate to verify that, for every small  $\epsilon > 0$ , the traffic pattern loading the switch is admissible.

We notice that, under these assumptions, high priority and low priority cells are never transferred at the same time. All  $i \rightarrow i$  cells, with  $0 \leq i < P - 1$ , are transferred together, while  $(P - 1) \rightarrow (P - 1)$  cells traverse the switch alone. Thus in

order to guarantee the full transfer of all the cells arriving at the switch, it must be  $(P - 1)q + p + q \leq S$ , i.e.,  $S \geq 2 - 1/P$ .

*Sufficiency.* It was proved in [5] that a CIOQ switch with speed-up  $2 - 1/P$  can exactly emulate an OQ switch operating on different traffic classes with a strict priority discipline (in the sense that cells can depart from the two systems at the same time). The proof follows immediately. ■

Speed-up  $2 - 1/P$  is sufficient, as proved in [5], to guarantee 100% throughput under any admissible traffic pattern for quite a large class of multiclass service disciplines. However, since the implementation cost of the scheme proposed in [5] can be significantly large, both in terms of internal bandwidth due to the required speed-up, and in terms of algorithmic complexity of the scheduler, the identification of *simpler* multiclass schedulers that allow a pure IQ to achieve good performance is fundamental.

In the rest of this section we focus on the definition of a wide class of schedulers that achieve 100% throughput in a multiclass traffic environment.

*Definition 10:* Let  $F(X)$  be a regular function<sup>4</sup>  $F \in C^1[\mathbb{R}^{+N} \rightarrow \mathbb{R}^{+N}]$ . An IQ switch adopts a  $F(X)$ -max-scalar scheduling policy if the selection of the switching vector in each slot is implemented according to the following rule:

$$D_n = \arg \left( \max_{D_i \in \mathcal{D}_{X_n}} F(X_n) D_i^T \right) \quad (4)$$

where  $X_n$  is the vector of queue lengths, and  $\mathcal{D}_{X_n}$  denotes the set of all possible switching vectors at time  $n$ .

*Theorem 2:* Let  $F(X)$  be a regular function  $F \in C^1[\mathbb{R}^{+N} \rightarrow \mathbb{R}^{+N}]$  such that:

1.  $F(X)$  defines a conservative field, i.e.:

$$\oint_{\Gamma} F(X) d\Gamma(X)^T = 0 \quad (5)$$

for each regular closed line  $\Gamma$  in  $\mathbb{R}^{+N}$

2.  $F(X)$  grows to infinity when  $X$  grows to infinity; formally, there exists a finite  $s > 0$  such that:

$$\liminf_{\|X\| \rightarrow \infty} \frac{\|F(X)\|}{\|X\|} \geq s \quad (6)$$

3. all null elements of  $X$  remain null:

$$\mathbb{I}[X]F(X) = F(X) \quad (7)$$

Then an IQ switch adopting the  $F(X)$ -max-scalar policy is strongly stable under any admissible i.i.d. traffic pattern.

*Proof:* Let us define the function  $\mathcal{L}(X)$ :

$$\mathcal{L}(X) = \int_{\Gamma_X} F(Y) d\Gamma_X(Y)^T \quad (8)$$

$$\mathcal{L}(0) = 0 \quad (9)$$

where  $\Gamma_X$  is an open regular line with endpoints 0 and  $X$ .

By definition  $\mathcal{L}(X) \in C^2[\mathbb{R}^{+N} \rightarrow \mathbb{R}]$ . It is easy to verify that, for each  $X \in \mathbb{R}^{+N}$ ,  $\mathcal{L}(X) \geq 0$ . To see this, it is sufficient

<sup>4</sup> $C^n$  denotes the set of continuous functions with continuous  $i$ -th derivative,  $1 \leq i \leq n$ .

to consider a straight line  $\Gamma_X$  parallel to vector  $X$ . Being  $X \in \mathbb{R}^{+N}$ , both  $F(Y)$  and  $d\Gamma_X(Y)$  in (8) belong to  $\mathbb{R}^{+N}$  for all  $Y$ , so that also  $\mathcal{L}(X) \in \mathbb{R}^{+N}$ .

Let us consider  $\mathcal{L}(X)$  as our Lyapunov function. Since the maximum number of cells arriving in a slot at the switch is bounded, then  $\|X_{n+1}\|_2$  is bounded for any finite  $X_n$ , and from the regularity of  $\mathcal{L}(X)$  follows that:

$$E[\mathcal{L}(X_{n+1}) | X_n] < \infty$$

Finally, for  $\|X_n\|_2 \rightarrow \infty$ , by writing a Taylor series for  $\mathcal{L}(X_n + A_n - D_n) = \mathcal{L}(X_n) + \nabla \mathcal{L}(X_n)(A_n - D_n)^T + \dots$ , we obtain:

$$\begin{aligned} \frac{E[\mathcal{L}(X_{n+1}) | X_n] - \mathcal{L}(X_n)}{\|X_n\|_2} &= O\left(\frac{\nabla \mathcal{L}(X_n)(E[A] - D_n)^T}{\|X_n\|_2}\right) \\ &= O\left(\frac{F(X_n)(E[A] - D_n)^T}{\|X_n\|_2}\right) \end{aligned} \quad (10)$$

We must now show that (10) is smaller than a negative finite constant. By the Birkoff-von-Neumann theorem [14], every vector  $Y$  in  $\mathbb{R}^{+N}$  such that  $\|Y\|_{IO} \leq 1$  belongs to the convex hull of the switching vectors. Since the arrival process is admissible, hence it is internal to the convex hull generated by departure vectors ( $\|A\|_{IO} < 1$ ), there exists an  $\epsilon > 0$ , and a vector  $A' = E[A] + \epsilon D_n$ ,  $A' \in \mathbb{R}^{+N}$ , which is again internal to the convex hull ( $\|A'\|_{IO} < 1$ ). We can write  $E[A] = A' - \epsilon D_n$ , and substitute in the right-hand side of (10), whose numerator becomes  $[F(X_n)(A' - \epsilon D_n - D_n)^T]$ . Now, by the linearity of functional  $F(X_n)Y^T$  with respect to  $Y^T$ , and the definition of  $F(X)$ -max-scalar policy, it follows that, under the assumptions of the theorem,  $F(X_n)A' \leq \max_{D^* \in \mathcal{D}_{X_n}} F(X_n)D^{*T} = \mathcal{H}[X_n]F(X_n)D_n^T$ , thus:

$$\frac{E[\mathcal{L}(X_{n+1})|X_n] - \mathcal{L}(X_n)}{\|X_n\|_2} \leq -\epsilon \frac{F(X_n)D_n^T}{\|X_n\|_2}$$

Then, for  $\|X_n\|_2$  growing to infinity, using (6) and the fact that  $\|D_n\|$  is always finite,

$$\frac{E[\mathcal{L}(X_{n+1})|X_n] - \mathcal{L}(X_n)}{\|X_n\|_2} < -\epsilon'$$

where  $\epsilon'$  is a positive constant depending on  $N$  and  $F(X)$ . ■

Note that Condition (6) of Theorem 2, while permitting to associate different finite weights with different traffic classes, prevents strict priorities among traffic classes, which would require infinite weight ratios.

The proof can be extended as follows to more general traffic processes, by relaxing the stability conditions.

*Theorem 3:* An IQ switch adopting the  $F(X)$ -max-scalar policy satisfying the conditions of Theorem 2, and such that  $F(\alpha X) = \alpha F(X)$  for all scalars  $\alpha$ , achieves 100% throughput under any admissible traffic pattern satisfying the strong law of large numbers.

*Proof:* Let us write the fluid equations for the switch in isolation:

$$\dot{X}(t) = \Lambda t - D(t) \quad (11)$$

The work-conserving, max-scalar service policy can be specified as follows (see [11])<sup>5</sup>:

$$\dot{D}(t) = \sum_{\alpha} \dot{w}_{\alpha}(t) \Pi_{\alpha} \mathcal{H}[X(t)]$$

<sup>5</sup>  $\dot{f}(t)$  denotes the derivative of  $f(t)$  with respect to time  $t$ .

where  $\Pi_{\alpha}$  is a switch permutation (from which the departure vector can be computed knowing the state of the queues), and  $\dot{w}_{\alpha}(t)$  is the instantaneous amount of work performed by permutation  $\Pi_{\alpha}$ . Subscript  $\alpha$  runs on all switching permutations. The work-conserving policy is specified by the constraint:

$$\sum_{\alpha} \dot{w}_{\alpha}(t) = 1$$

while the max-scalar policy can be described as:

$$\dot{w}_{\alpha}(t) = 0 \text{ if } \exists \alpha' : F(X(t))\Pi_{\alpha'}^T > F(X(t))\Pi_{\alpha}^T$$

Finally, the following equation represents in the fluid model Property (6):

$$\liminf_{\|X\| \rightarrow \infty} \frac{\|F(X(t))\|}{\|X(t)\|} \geq s > 0 \quad (12)$$

Defining the Lyapunov function  $\mathcal{L}(X) = \int_{\Gamma_X} F(Y)d\Gamma_X(Y)^T$ , where  $\Gamma_X$  is a regular line in  $\mathbb{R}^{+N}$  whose endpoints are 0 and  $X$ , it is easy to verify, with the same arguments of the previous proof, that  $\mathcal{L}(X)$  is null for  $X = 0$  and greater than 0 for each  $X \neq 0$ . Moreover it results:

$$\dot{\mathcal{L}}(X(t)) < 0$$

whenever  $X(t) \neq 0$ . Indeed, by writing the derivative of  $\mathcal{L}(X(t))$ ,

$$\dot{\mathcal{L}}(X(t)) = \nabla \mathcal{L}(X(t))\dot{X}(t) = F(X(t))[\Lambda - \dot{D}(t)]^T$$

However, since  $\Lambda$  is in the convex hull of the switching vectors  $\dot{D}(t)$ , by the definition of the  $F(X)$ -max-scalar policy, and by the linearity of function  $F(X(t))[\Lambda - \dot{D}(t)]^T$  with respect to  $D(t)$ , it follows:

$$\dot{\mathcal{L}}(X(t)) = F(X(t))[\Lambda - \dot{D}(t)]^T < 0 \quad \forall X(t) \neq 0$$

Thus:

$$\dot{\mathcal{L}}(X(t)) < 0 \quad \text{whenever } \mathcal{L}(X(t)) \geq 0$$

As a conclusion<sup>6</sup>,  $X(t) = 0$  is the only solution to the fluid equation (11) under the initial condition  $X(0) = 0$ , and the system of queues is rate stable under any admissible traffic pattern. ■

It can be easily verified that, for each symmetric copositive matrix  $W$ ,  $F(X) = XW$  (note that  $F(X)$  is now a function in  $C^{\infty}[\mathbb{R}^{+N} \rightarrow \mathbb{R}^{+N}]$ ) satisfies Properties (5) and (6). Note that  $F(X)$  can be seen as the gradient of function  $\mathcal{L}(X) = \frac{1}{2}XWX^T$ . To meet also constraint (7), we take  $W$  to be diagonal, and state the following result.

*Corollary 1:* Let  $W$  be a diagonal copositive matrix, and let  $F(X)$  be a function in  $C^{\infty}[\mathbb{R}^{+N} \rightarrow \mathbb{R}^{+N}]$  defined as  $F(X) = XW$ . Then a multiclass switch implementing the  $F(X)$ -max-scalar policy is strongly stable under any admissible traffic pattern if the number of arrivals at VOQs in each slot forms an i.i.d. sequence. The switch is rate stable, under any admissible traffic pattern, if the sequences of arrivals at the VOQs satisfy the strong law of large numbers.

<sup>6</sup> Assume  $\mathcal{L}(t) \geq 0$ ,  $\mathcal{L}(0) = 0$ , and  $\dot{\mathcal{L}}(t) \leq 0$ . Consider  $\mathcal{L}^2(t) = 2 \int_0^t \mathcal{L}(x) d\mathcal{L}(x)$ . By definition  $\mathcal{L}^2(t) \geq 0$ ; but  $\int_0^t \mathcal{L}(x) d\mathcal{L}(x) \leq 0$ . Hence  $\mathcal{L}(t) = 0$ ,  $\forall t$  (see [11]).

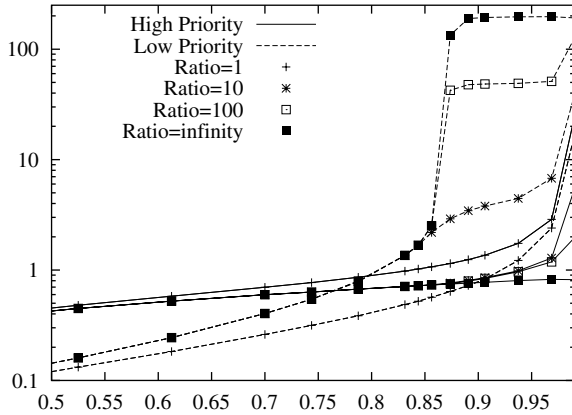


Fig. 3. Average queue occupancies versus offered load for different MWM policies serving with different weight ratios a two-class asymmetric traffic pattern.

### C. Implementation Issues and Simulation Results

The implementation of the  $F(X)$ -max-scalar-policy requires the solution of a maximum weight matching (MWM) problem, which is well known in graph theory, and the best known solutions [22] have a complexity  $O(N^3)$ . Several approximate solutions were proposed [2], [24], [12], [23], reducing the complexity to  $O(N^2)$  or less.

The results of Theorems 2 and 3 state that different traffic classes can be handled, and that it is possible to serve with different rates the different classes, but it is not possible to have strict priorities among traffic classes, because it is always necessary to respond to queues growing to infinite occupancy, if 100 % throughput is wanted.

We show in this section simulation results that confirm these statements. We consider a  $8 \times 8$  IQS in isolation loaded by a multiclass traffic pattern similar to the one presented in Section III-B. Inputs  $0 \leq i \leq 6$  send high-priority traffic with rate  $7x$  to output  $i$ . Inputs  $0 \leq i \leq 6$  send low-priority traffic with rate  $x$  to output 7, and input 7 sends low-priority traffic with rate  $x$  to outputs  $0 \leq i \leq 6$ . Rate  $x$  is varied from 0 to  $\frac{1}{8}$  in order to have different port loads. VOQs can store up to 200 cells. We only consider MWM policies, with four types of  $F(X)$  functions: equal weight to the two priorities (corresponding to a single traffic class), weight ratios 10 and 100 between high and low priority, and strict priority (corresponding to an infinite weight ratio). Fig. 3 shows average queue occupancies observed by simulation for high and low priority traffic versus the total offered load. Losses (not reported in the figure) were observed only for the strict priority case, starting from an average offered load equal to approximately  $\frac{7}{8}$ . When the weight ratio is 100, low priority cells face large delays (but no losses), since the queue has to grow large enough to be taken into proper account by the  $F(X)$ -max-scalar policy. The increased queue occupancies can be well observed for offered loads between  $\frac{7}{8}$  and 1. Note that a good differentiation between the two traffic classes is obtained by assigning different weights without paying the cost of throughput limitations.

## IV. NETWORKS OF SWITCHES

### A. Notation

We consider in this section a network of  $K$  input-queueing cell-based switches. Switch  $k$ ,  $0 \leq k < K$ , has  $P_k$  input ports and  $P_k$  output ports, all at the same cell rate. Each switch handles  $C$  classes of traffic, and performs a MCVOQ at inputs. Thus there are  $CP_k^2$  different VOQs at switch  $k$ .

The network of switches can be thus modelled as a system  $Q$  containing  $N = \sum_k CP_k^2$  virtual queues. We restrict our study to the case  $P_k = P \forall k$ , so that  $N = CP^2K$ . Let  $S(n)$  be the function that returns the switch on which VOQ  $n$  is located; let  $I(n)$  be the function that returns the index of the input card at switch  $S(n)$  on which the VOQ is located; let  $O(n)$  be the function that returns the index of the output card at switch  $S(n)$  to which VOQ cells are directed; let finally  $C(n)$  be the function that returns the index of the traffic class associated with queue  $n$ . The queue at input  $I(n)$  of switch  $S(n)$  storing cells of class  $C(n)$  directed to output  $O(n)$  is called  $q^{(n)}$ .

We adapt as follows the concept of  $\|Z\|_{\max L}$  to the case of a network of switches handling multiclass traffic.

*Definition 11:* Given a vector  $Z \in \mathbb{R}^N$ ,  $Z = \{z^{(n)}, n = CP^2k + CPi + Cj + l, 0 \leq k < K, i, j = 0, 1, \dots, P-1, l = 0, 1, \dots, C-1\}$ , the norm  $\|Z\|_{IO}$  is defined as:

$$\|Z\|_{IO} = \max_{\substack{k=0, \dots, K-1 \\ i=0, \dots, P-1}} \left\{ \sum_{n \in S^{-1}(k) \cap O^{-1}(i)} |z^{(n)}|, \right. \\ \left. \sum_{n \in S^{-1}(k) \cap I^{-1}(i)} |z^{(n)}| \right\}$$

At each time slot, a set of non contending cells departs from the VOQ of each switch. More formally, we say that:

*Definition 12:* At each time slot, the departure vector  $D \in \{0, 1\}^N$  satisfies:

$$\|D\|_{IO} \leq 1$$

The  $N \times N$  matrix  $R_n = [r_n^{(k,l)}]$  is the routing matrix of the network, whose element  $r_n^{(k,l)}$  represents the fraction of customers departing from VOQ  $k$  in time interval  $(n, n+1]$  that enter VOQ  $l$ .

*Definition 13:* The traffic pattern loading a network of IQSs is admissible if and only if

$$\|E\|_{IO} = \|\Lambda(I - R)^{-1}\|_{IO} < 1$$

where  $R = E[R_n]$  was defined in Sect. II-A.

Note that an admissible traffic pattern can be transferred without losses in a network of output buffered switches.

### B. Main Results for a Network of Switches

In [19] it has been shown that a particular network of IQ switches exhibits an unstable behavior under admissible traffic patterns, even when the switches implement a policy that would guarantee the stability of each switch in isolation under the same load. In this section we try to formalize and generalize such result by providing a general definition of the stability region of a network of IQ switches obtained through the fluid models theory.

Let us introduce our first result.

*Theorem 4:* An open network of multiclass switches implementing the  $F(X)$ -max-scalar policy is rate stable under each admissible traffic pattern such that arrival sequences at VOQs satisfy the strong law of large numbers, if:

- $G(X) = F(X)[(I - R)^{-1}]^T$  defines a conservative field;
- $F(X)$  satisfies conditions (6) and (7);
- $F(\alpha X) = \alpha F(X)$  for all scalars  $\alpha$ .

*Proof:* Let us write the fluid equations:

$$X(t) = \Lambda t - D(t)(I - R)$$

with the constraints for a max-scalar service policy:

$$\begin{aligned} \dot{D}(t) &= \sum_{\alpha} \dot{w}_{\alpha}(t) \Pi_{\alpha} \mathbb{I}[X(t)] \\ \sum_{\alpha} \dot{w}_{\alpha}(t) &= 1 \end{aligned}$$

$$\dot{w}_{\alpha}(t) = 0 \text{ if } \exists \alpha' : F(X(t)) \Pi_{\alpha'}^T > F(X(t)) \Pi_{\alpha}^T$$

Note that the two expressions above that define the  $F(X)$ -max-scalar policy are equivalent to:

$$\dot{D}(t) = \arg \max_{\Pi_{\alpha}} F(X(t)) \Pi_{\alpha}^T$$

Being the network of switches open, and being  $F(X)[(I - R)^{-1}]^T$  a conservative field,  $\sum_{i=0}^{\infty} R^i$  converges to the finite copositive matrix  $(I - R)^{-1}$ . We define as Lyapunov function of the system:

$$\mathcal{L}(X) = \int_{\Gamma_X} F(Y)[(I - R)^{-1}]^T d\Gamma_X^T(Y)$$

where  $\Gamma_X$  is a regular line whose endpoints are 0 and  $X$ .

Since  $(I - R)^{-1} - I = \sum_{i=1}^{\infty} R^i$  is weakly copositive<sup>7</sup>, it results  $\mathcal{L}(X) \geq \int_{\Gamma_X} F(Y) d\Gamma_X^T(Y) > 0 \quad \forall X \neq 0$ . Let us write the time derivative of  $\mathcal{L}(X(t))$ :

$$\dot{\mathcal{L}}(X(t)) = \nabla \mathcal{L}(X(t)) \dot{X}(t)^T = F(X(t))[(I - R)^{-1}]^T \dot{X}(t)^T$$

Substituting in the relation above the expression of  $\dot{X}(t) = \Lambda - \dot{D}(t)(I - R)$ , we obtain:

$$\dot{\mathcal{L}}(X(t)) = F(X(t))[(I - R)^{-1}]^T [\Lambda - \dot{D}(t)(I - R)]^T$$

Then:

$$\begin{aligned} \dot{\mathcal{L}}(X(t)) &= F(X(t))[(I - R)^{-1}]^T \Lambda^T - F(X(t)) \dot{D}(t)^T \\ &= F(X(t))[(I - R)^{-1}]^T \Lambda^T - \\ &\quad - F(X(t)) \left[ \sum_{\alpha} \dot{w}_{\alpha}(t) \Pi_{\alpha} \mathbb{I}[X(t)] \right]^T = \\ &= F(X(t))[(I - R)^{-1}]^T \Lambda^T - \\ &\quad - F(X(t)) \left( \sum_{\alpha} \dot{w}_{\alpha}(t) \Pi_{\alpha} \right)^T = \\ &= F(X(t))[(I - R)^{-1}]^T \Lambda^T - \\ &\quad - F(X(t)) \left( \arg \max_{\Pi_{\alpha}} \Pi_{\alpha} F(X(t))^T \right)^T \quad (13) \end{aligned}$$

<sup>7</sup>Matrix  $W \in \mathbb{R}^{+IN} \times \mathbb{R}^{+IN}$  is said weakly copositive if, for each  $X \in \mathbb{R}^{+IN}$ ,  $XW X^T$  is non-negative.

By definition of the  $F(X)$ -max-scalar policy, for each  $\Lambda$  such that  $\Lambda(I - R)^{-1}$  belongs to the convex hull of  $\Pi_{\alpha}$ , expression (13) is negative. Thus for each traffic pattern such that  $\|\Lambda(I - R)^{-1}\|_{IO} < 1$  the network of switches is rate stable. ■

A similar theorem, not reported here for space limitations, can be proven, using Lyapunov functions, for admissible i.i.d. arrival processes without requiring the condition  $\alpha F(X) = F(\alpha X)$ .

The problem of the existence of a scheduling policy for interconnected IQ switches that makes the network rate stable under any admissible traffic pattern is then related to the existence of a function  $F(X)$  at each switch such that (6) and (7) are satisfied, and in addition  $F(X)[(I - R)^{-1}]^T$  defines a conservative field.

Note that  $F(X) = X M (I - R)^T$ , where  $M$  is a copositive diagonal matrix, satisfies both (6) and (7), and

$$F(X)[(I - R)^{-1}]^T = X M (I - R)^T [(I - R)^{-1}]^T = X M$$

hence  $F(X)[(I - R)^{-1}]^T$  defines a conservative field. The policy  $F(X) = X M (I - R)^T$  can be implemented in a distributed fashion by running a ‘‘local’’ MWM algorithm at each switch. This requires to associate with local virtual queues weights of the form  $\sum_{i=0}^{N-1} x^{(i)} M^{(i,i)} r^{(i,j)}$ , i.e., it requires the knowledge of the lengths of VOQs at neighboring switches (switches that can be directly reached from the considered switch). Thus some form of interaction (through signalling) is required among the switches of the network.

Note also that this optimal policy cannot be exactly implemented in a network of switches due to the delay between switches. It can however be approximately implemented by acquiring at each switch an approximate knowledge of the queues state at neighboring switches. Note that only the information about the length of few queues must be periodically exchanged among each pair of neighboring nodes.

In general, when  $F(X)[(I - R)^{-1}]^T$  does not form a conservative field, Theorem 4 provides no insight on the stability region of the network of switches. The methodology developed in the proof of the theorem, can be however extended to find conditions on the stability region of policies that do not define a conservative field. Indeed, restricting our analysis to  $F(X) = X M$  policies, whenever  $M[(I - R)^{-1}]^T$  is not symmetric, and thus does not define a conservative field, it is always possible to find a matrix  $B$ , such that  $M[(I - R)^{-1}]^T + B$  results symmetric; by defining as Lyapunov function of the system:

$$\mathcal{L}(X) = \frac{1}{2} X \{ M[(I - R)^{-1}]^T + B \} X^T$$

it results:

$$\begin{aligned} \dot{\mathcal{L}}(X(t)) &= X(t) \{ M[(I - R)^{-1}]^T + B \} \dot{X}(t)^T = \\ &= X(t) \{ M[(I - R)^{-1}]^T + B \} [\Lambda - \dot{D}(t)(I - R)]^T \\ &= X(t) \{ M[(I - R)^{-1}]^T + B \} \Lambda^T - \\ &\quad - X(t) M \dot{D}(t)^T - X(t) B (I - R)^T \dot{D}(t)^T = \\ &= X(t) \{ M[(I - R)^{-1}]^T + B \} \Lambda^T - \\ &\quad - X(t) M \Pi_F^T - X(t) B (I - R)^T \Pi_F^T = \\ &= X(t) M \{ [(I - R)^{-1}]^T + M^{-1} B \} \Lambda^T - \\ &\quad - M^{-1} B (I - R)^T \Pi_F^T \} - X(t) M \Pi_F^T \end{aligned}$$

where  $\Pi_F$  is the switching matrix selected according to the  $F(X)$ -max-scalar policy. Thus the policy is rate stable for each  $\Lambda$  when the term between braces belongs to the convex hull defined by departure vectors, i.e. when:

$$\|[(I - R)^{-1}]^T + M^{-1}B\}\Lambda^T - M^{-1}B(I - R)^T \Pi_F^T\|_{IO} < 1$$

Note that, since  $\Pi_F$  depends on  $X(t)$ , and in general can be any switching matrix, the above inequality must be satisfied for any permutation matrix. Note, finally, that the satisfaction of the equation above represents a sufficient (but not necessary) condition for stability.

We now introduce another result.

*Theorem 5:* An open network of multiclass IQS implementing the  $F(X)$ -max-scalar policy, with  $F(X) = XW(I - R)^{-1}$ , being  $W$  a diagonal copositive matrix, is rate stable under each admissible traffic pattern such that the sequences of arrivals at VOQs satisfy the strong law of large numbers.

*Proof:* Let us consider the Lyapunov function:

$$\mathcal{L}(X) = \frac{1}{2} X(I - R)^{-1} [(I - R)^{-1}]^T W X^T$$

It is immediate to see that  $\mathcal{L}(X) > 0 \forall X \neq 0$ , and  $\mathcal{L}(0) = 0$ . The derivative of  $\mathcal{L}(X(t))$  is:

$$\begin{aligned} \dot{\mathcal{L}}(X(t)) &= \dot{X}(t)(I - R)^{-1} [(I - R)^{-1}]^T W X^T(t) = \\ &= [\Lambda - \dot{D}(t)(I - R)](I - R)^{-1} [(I - R)^{-1}]^T W X^T(t) \\ &= \Lambda(I - R)^{-1} [(I - R)^{-1}]^T W X^T(t) - \\ &\quad - \dot{D}(t) [(I - R)^{-1}]^T W X^T(t) \end{aligned}$$

Where the last expression is negative for each admissible vector  $\Lambda$ , i.e.  $\Lambda$  such that  $\|\Lambda(I - R)^{-1}\|_{IO} < 1$  ■

Also in this case, the implementation of the  $F(X)$ -max-scalar policy can be performed in a distributed fashion; however, in this case, the implementation requires an exchange of information among all the switches that are crossed by the same flow.

### C. Simulation Results

We consider the network of eight IQSs depicted in Fig. 4, in which continuous lines represent links between switches, and dashed lines represent information flows and their routing in the network. Note that each pair of IQSs (all pairs are alike) is traversed by a locally originated flow, a locally terminating flow, and an in-transit flow. The cell arrival process at the source of each flow is Bernoulli, and the arrival rate for each flow is 0.33 times the link data rate. In-transit and terminating flows are given weight 10 times larger than locally originating flows.

Fig. 5 shows that queue lengths take a divergent oscillating behavior when a local MWM scheduling is adopted, while they remain finite when the scheduling accounts for the state of adjacent switches, as stated in Theorem 4 (note the different vertical scales).

## V. CONCLUSIONS

We considered in this paper input-queued packet switches under multiclass traffic. Two are our most important results:

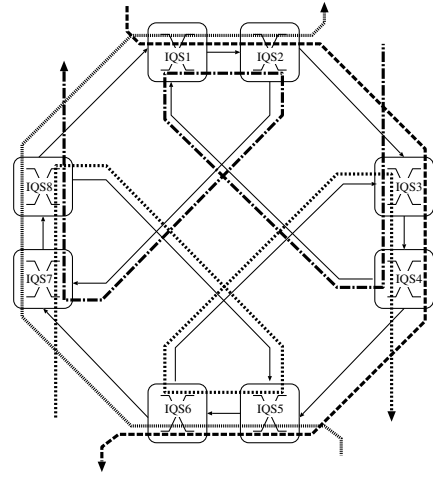


Fig. 4. The network of IQSs considered in our simulation.

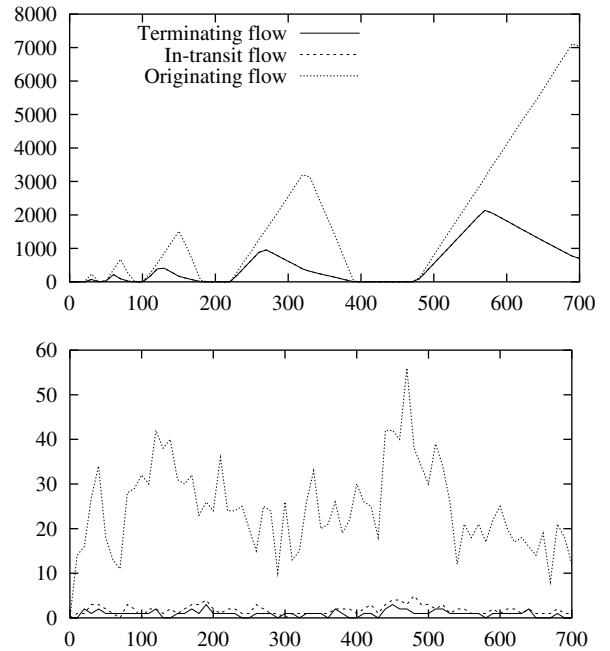


Fig. 5. Queue lengths versus time (in thousands of slots) for the three flows at IQS1, when a local MWM scheduling policy (upper plot), and a “distributed” MWM according to the requirements of Theorem 4 (lower plot), are used.

- we defined a large class of scheduling algorithms, called  $F(X)$ -max-scalar, that guarantee stability to a switch in isolation under admissible multiclass traffic patterns;
- we extended the above result to networks of interconnected switches, showing that state information must be exchanged among adjacent switches to guarantee stability.

These important and rather general results were obtained analytically, using both the Lyapunov function methodology and the fluid models approach, and were validated with simple simulation experiments. Given the ever-increasing thirst for channel bandwidth and for switching capacity, our results can find an immediate application in the design of high-speed packet networking infrastructures. They may end-up highlighting unexpected behaviors when priority-based scheduling disciplines are imple-

mented, as it is the case for example in the DiffServ approach to QoS defined by the IETF.

[25] H.J.Kushner, *Stochastic Stability and Control*, Academic Press, 1967.

## REFERENCES

- [1] T.Anderson, S.Owicki, J.Saxe, C.Thacker, "High Speed Switch scheduling for local area networks", *ACM Transactions on Computer Systems*, Nov.1993, pp. 319-352.
- [2] N.McKeown, *Scheduling algorithms for input-queued cell switches*, Ph.D. Thesis, Un. of California at Berkeley, 1995.
- [3] H.Zhang, "Service disciplines for guaranteed performance service in packet-switching networks", *Proceedings of the IEEE*, vol.83, n.10, Oct.1995, pp.1374-1399.
- [4] D.Stiliadis, A.Varma, "Providing bandwidth guarantees in an input-buffered crossbar switch", *IEEE INFOCOM 95*, Boston, MA, USA, Apr.1995, pp.960-968.
- [5] S.T.Chuang, A.Goel, N.McKeown, B.Prabhakar, "Matching output queuing with combined input and output queuing", *IEEE Journal on Selected Areas in Communications*, vol.17, n.6, Dec.1999, pp.1030-1039.
- [6] N.McKeown, A.Mekkittikul, V.Anantharam, J.Walrand, "Achieving 100% throughput in an input-queued switch", *IEEE Transactions on Communications*, vol.47, n.8, Aug.1999, pp. 1260-1272.
- [7] E.Leonardi, M.Mellia, F.Neri, M.Ajmone Marsan, "On the Stability of Input-Queued Switches with Speedup", *IEEE/ACM Transactions on Networking*, vol.9, n.1, Feb.2001, pp.104-118.
- [8] N.McKeown, A.Mekkittikul, "A practical scheduling algorithm to achieve 100% throughput in input-queued switches", *IEEE INFOCOM 98*, San Francisco, CA, USA, Apr.1998, pp.792-799.
- [9] J.G.Dai, "Stability of Fluid and Stochastic Processing Networks", Miscellaneous Publication n.9, Centre for Mathematical Physics and Stochastic, Denmark (<http://www.maphysto.dk>), Jan.1999.
- [10] J.G.Dai, "On Positive Harris Recurrence of Multiclass Queueing Networks: a Unified Approach Via Fluid Limit Models", *Annals of Applied Probability*, n.5, pp.49-77, 1995.
- [11] J.G.Dai, B.Prabhakar, "The throughput of data switches with and without speedup", *IEEE INFOCOM 2000*, Tel Aviv, Israel, Mar.2000, pp.556-564.
- [12] M.Ajmone Marsan, A.Bianco, E.Leonardi, L.Milia, "RPA: a flexible scheduling algorithm for input buffered switches", *IEEE Transactions on Communications*, vol.47, n.12, Dec.1999, pp.1921-1933.
- [13] A.Hung, G.Kesidis, N.McKeown, "ATM input-buffered switches with the guaranteed-rate property", *ISCC '98*, Athens, Greece, June 1998, pp.331-335.
- [14] C.Cheng-Shang, C. Wen-Jyh, H. Hsiang-Yi, "Birkhoff-von Neumann input buffered crossbar switches", *INFOCOM 2000*, Tel Aviv, Israel, Apr.2000, pp.1614-1623.
- [15] V.Tabatabaee, L.Georgiadis, L.Tassiulas, "QoS provisioning and tracking fluid policies in input queueing switches", *INFOCOM 2000*, Tel Aviv, Israel, Apr.2000, pp.1624-1633.
- [16] P. Krishna, N.S.Patel, A.Charny, R.J.Simcoe, "On the speedup required for work-conserving crossbar switches", *IEEE Journal on Selected Areas in Communications*, vol.17, n.6, June 1999, pp.1057-1066.
- [17] A.C.Kam, Kai-Yeung Siu, "Linear-complexity algorithms for QoS support in input-queued switches with no speedup", *IEEE Journal on Selected Areas in Communications*, vol.17, n.6, June 1999, pp.1040-1056.
- [18] F.M.Chiusi, A.Francini, "Providing QoS guarantees in packet switches", *GLOBECOM 1999*, Rio de Janeiro, (Brazil), Dec.1999, pp.1307-1312.
- [19] M.Andrews, L.Zhang, "Achieving stability in networks of input-queued switches", *INFOCOM 2001*, Anchorage, Alaska, Apr.2001, pp.1673-1679.
- [20] E.Leonardi, M.Mellia, M.Ajmone Marsan, F.Neri, "On the Throughput Achievable by Isolated and Interconnected Input-Queueing Switches under Multiclass Traffic", Technical Report, Dipartimento di Elettronica, Politecnico di Torino, July 2001, <http://www.tlc-networks.polito.it/~neri/IQS-nets.ps>.
- [21] F.Baskett, K.M.Chandy, R.R.Muntz, F.Palacios, "Open, closed and mixed networks with different classes of customers", *Journal of the ACM*, vol.22, n.2, April 1975, pp.248-260.
- [22] R.E.Tarjan, *Data structures and network algorithms*, Society for Industrial and Applied Mathematics, Pennsylvania, Nov.1983.
- [23] L.Tassiulas, "Linear complexity algorithms for maximum throughput in radio networks and input queued switches", *IEEE INFOCOM 98*, San Francisco, CA, USA, Apr.1998, pp.553-559.
- [24] V.Tabatabaee, L.Georgiadis, L.Tassiulas, "QoS provisioning and tracking fluid policies in input queueing switches", *INFOCOM 2000*, Tel Aviv, Israel, Apr.2000, pp.1624-1633.
- [24] R.O.LaMaire, D.N.Serpanos, "Two dimensional round-robin schedulers for packet switches with multiple input queues", *IEEE/ACM Transactions on Networking*, vol.2, n.5, Oct.1994, pp. 471-482.