
Theoretical Performance of Input Queued Switches using Lyapunov Methodology

Andrea Bianco¹, Paolo Giaccone¹, Emilio Leonardi¹, Marco Mellia¹, and Fabio Neri¹

Dipartimento di Elettronica, Politecnico di Torino, C.so Duca degli Abruzzi 24, Torino, Italy
firstname.lastname@polito.it

Summary. The stochastic Lyapunov methodology is a powerful analytical tool to assess the performance of queueing systems. We discuss here the main results obtained applying this methodology to the context of packet switch architectures with input queues, considering both an isolated switch and networks of switches. The methodology allows to assess the limit performance in terms of throughput and delay; as a consequence, it is precious to devise optimal scheduling algorithms. The theoretical results presented here can inspire practical design of high-speed packet switches.

1 Introduction

In recent years a great attention has been devoted by the research community to the design of Input Queued (IQ) packet switching architectures and to the assessment of their performance.

An IQ switch architecture, depicted in fig. 1, is usually built around any non-blocking bufferless switching fabric, interconnecting input to output lines; examples of such fabric are crossbars, Clos networks, Benes networks, Batcher-Banyan networks. At any time the switching fabric can be configured to provide a set of parallel input/output connections, later called “matching”, such that each input (output) is

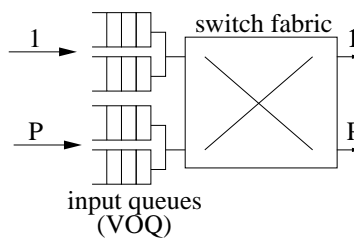


Fig. 1. Basic architecture of a $P \times P$ IQ switch.

connected with at most one output (input). All switching fabric connections run at the same speed of external lines, (normally assumed to be equal for all inputs and outputs); indeed, this allows the design of high-speed switching fabrics. Buffering and local processing is available for each external line in a line card, which provides termination of the line and interfacing to the fabric. In case of contention, packets are stored in buffers at the input line cards. IQ switches have become an attractive architectural solution for the design of large-size and high-capacity packet switches since the seminal works [4, 23, 30, 31] showed that the negative effects on performance of Head-of-the-Line (HoL) blocking [16] can be reduced or completely eliminated by adopting per-destination queueing (also called Virtual Output Queueing - VOQ) at input cards.

A major issue in the design of IQ packet switches is that the access to the switching fabric must be controlled by a scheduling algorithm, 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 significant processing power must be devoted to the scheduling algorithm, either at a centralized scheduler, or at line cards in a distributed manner.

We refer in this chapter to the case of fixed-size data units, called “cells”, 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 time (called “slot”) for all ports. Cell-based designs have been quite popular, as they permit to reduce the complexity both for the hardware architecture, and for the scheduling algorithm.

The problem faced by scheduling algorithms with VOQ can be formalized as a maximum size or maximum weight matching on a 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. If P is the number of ports, then the total number of possible switching configurations (matchings) is $P!$, corresponding to the number of input/output permutations.

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. But simpler algorithm may exhibit reduced performance. Hence, a possible solution is to introduce a moderate speedup with respect to the data rate of input/output lines [9] in the switching fabric connections, 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 speedup is such that the internal switch bandwidth equals the sum of the data rates on input lines, input buffers become useless; in this case, the architecture becomes purely Output Queued (OQ).

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 analytical methodologies to assess the performance achievable by IQ and CIOQ switch architectures. A complete set of general theoretical results could indeed provide an important framework to drive

applied researchers toward better performing solutions. To this end, the stochastic Lyapunov function methodology played a fundamental role, since it permitted to obtain most of the known theoretical results about the throughput of IQ and CIOQ switches. In addition, the methodology was applied in the wider scenario of network of switches, providing new insights on their performance.

In this chapter we first briefly introduce the stochastic Lyapunov function methodology, showing how it can be successfully applied to determine the stability region of a system of queues; then, we summarize the main results about throughput performance of IQ and CIOQ architectures. We also show how the stochastic Lyapunov function methodology can be successfully applied to obtain bounds on the packet delay in IQ and CIOQ switches. Finally, we discuss the main results obtained for network of switches.

2 Theoretical framework

We introduce the theoretical framework to describe all the results regarding both switches in isolation and networks of switches, starting from the notation of a generic queueing system.

2.1 Description of the queueing system

Consider a system of J discrete-time queues (of infinite capacity) represented by¹ 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 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; it holds $N \geq J$. Let $L(k) = j$ be 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 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 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 timeslot. 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.

¹ All vectors are defined as row vectors.

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 of 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 $A_n = E_n$.

For simplicity of notation, we assume static routing (the extension to the case of dynamic routing is presented in [3]). The $N \times N$ matrix $R = [r^{(k,l)}]$ is the *routing matrix*, whose binary element $r^{(k,l)} = 1$ iff a customer served at virtual queue k is moved to virtual queue l . We assume that the system of queues forms an *open network*, i.e.,² $I + R + R^2 + R^3 + \dots = (I - R)^{-1}$ exists and is finite, i.e., $I - R$ is invertible. Note that $T_n = D_n R$. The law of evolution of virtual queues can thus be rewritten as:

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

Let us consider the external arrival 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 $[n, n+1)$.

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

Before proceeding, we recall some norms functions that will be helpful in the sequel.³

Definition 1 Given a vector $Z \in \mathbb{R}^N$, $Z = (z^{(k)}, 0 \leq k < N)$, norm $\|Z\|_p$ is defined as

$$\|Z\|_p = \left(\sum_{k=0}^{N-1} |z^{(k)}|^p \right)^{1/p}$$

² $E[X]$ denotes the expectation of random quantity X . I denotes the identity matrix, whose elements are equal to 1 on the diagonal, and null everywhere else.

³ Here, \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.

Definition 2 Given a location function $L(k) = j$, from $0 \leq k < N$ to $0 \leq j < J$, with $J \leq N$, norm $\|Z\|_{\max L}$ (the name refers to maximum queue length) is defined as

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

2.2 Stability definitions for a queueing system

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

Definition 3 Under a stationary exogenous arrival process $\{A_n\}$ satisfying the strong law of large numbers, i.e.:

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

A system of queues is rate stable 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 vector of queue sizes at time n .

Definition 4 Under a stationary exogenous arrival process $\{A_n\}$, 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 5 Under a stationary exogenous arrival process $\{A_n\}$, 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 rate stability. Indeed, the rate stability property allows queue sizes to indefinitely grow with sub-linear rate, while the weak stability property entails that the servers in the system of queues process the whole offered load, but the delay experienced by customers can be unbounded. Strong stability implies, in addition, the boundness of average queue sizes and customer delays.

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 [7]. This condition can be formalized as:

$$\|A(I - R)^{-1}\|_{\max L} < 1 \quad (4)$$

In general, as shown in [6, 10, 11], this condition does not guarantee the stability of a generic network of queues. Although condition (4) can be extended to

$$\|A(I - R)^{-1}\|_{\max L} \leq 1$$

when rate stability is considered, we will normally refer to the stricter formulation (4) here.

2.3 Lyapunov methodology

The systems under study can be modeled by discrete-time queues and they can be described with Discrete-Time Markov Chain (DTMC) models. Hence, we assume that the process describing the evolution of the system of queues is an irreducible DTMC, whose state vector at time n is $Y_n = (X_n, K_n)$, $Y_n \in \mathcal{I}^M$, $X_n \in \mathcal{I}^N$, $K_n \in \mathcal{I}^{N'}$, and $M = N + N'$. Y_n is the combination of vector X_n and a vector K_n of N' integer parameters. Let H be the state space of the DTMC, obtained as a subset of the Cartesian product of the state space H_X of X_n and the state space H_K of K_n .

From Definition 4, we can immediately see that if all states Y_n are positive recurrent, the system of queues is weakly stable; however, the converse is generally not true, since queue sizes can remain finite even if the states of the DTMC are not positive recurrent due to instability in the sequence of parameter $\{K_n\}$.

The following general criterion for the (weak) stability of systems that can be described with a DTMC is useful in the design of scheduling algorithms. This theorem is a straightforward extension of Foster's Criterion; see [13, 19, 33].

Theorem 1. *Given a system of queues whose evolution is described by a DTMC with state vector $Y_n \in \mathcal{I}^M$, if a lower bounded function $V(Y_n)$, called Lyapunov function, $V : \mathcal{I}^M \rightarrow \mathbb{R}$ can be found such that⁴ $E[V(Y_{n+1}) | Y_n] < \infty$, $\forall Y_n$, and there exist $\epsilon \in \mathbb{R}^+$ and $B \in \mathbb{R}^+$ such that $\forall \|Y_n\| > B$*

$$E[V(Y_{n+1}) - V(Y_n) | Y_n] < -\epsilon \quad (5)$$

then all states of the DTMC are positive recurrent and the system of queues is weakly stable.

⁴ We use the elementary notation for the conditional expectation, i.e., $E[X | Y \in A] = E[X, Y] / P\{A\}$, where A is an event set.

Note that an explicit dependence of the Lyapunov function on the time index n is allowed, so that it is possible to explicitly write $V(Y_n) = V(Y_n, n)$.

If the state space H of the DTMC is a subset of the Cartesian product of the denumerable state space H_X and a *finite* state space H_K , the stability criterion can be slightly modified, since the stability of the system can be inferred only from the queue size state vector X_n .

Corollary 1. *Given a system of queues whose evolution is described by a DTMC with state vector $Y_n \in \mathbb{N}^M$, and whose state space H is a subset of the Cartesian product of a denumerable state space H_X and a finite state space H_K , then, if a lower bounded function $V(X_n)$, called Lyapunov function, $V : \mathbb{N}^N \rightarrow \mathbb{R}$ can be found such that $E[V(X_{n+1}) | Y_n] < \infty \quad \forall Y_n$ and there exist $\epsilon \in \mathbb{R}^+$ and $B \in \mathbb{R}^+$ such that $\forall Y_n : \|X_n\| > B$*

$$E[V(X_{n+1}) - V(X_n) | Y_n] < -\epsilon \quad (6)$$

then all states of the DTMC are positive recurrent.

In this case, the system of discrete-time queues is weakly stable iff all states of the DTMC are positive recurrent.

In the following, we restrict our analysis to systems of queues for which Corollary 1 applies. The following criterion for *strong* stability extends the previous result:

Theorem 2. *Given a system of queues whose evolution is described by a DTMC with state vector $Y_n \in \mathbb{N}^M$, and whose state space H is a subset of the Cartesian product of a denumerable state space H_X and a finite state space H_K , then, if a lower bounded function $V(X_n)$, called Lyapunov function, $V : \mathbb{N}^N \rightarrow \mathbb{R}$ can be found such that $E[V(X_{n+1}) | Y_n] < \infty \quad \forall Y_n$ and there exist $\epsilon \in \mathbb{R}^+$ and $B \in \mathbb{R}^+$ such that $\forall Y_n : \|X_n\| > B$*

$$E[V(X_{n+1}) - V(X_n) | Y_n] < -\epsilon \|X_n\| \quad (7)$$

then the system of queues is strongly stable.

We report here the proof of the theorem, derived by [21]. This proof can be useful for a reader interested in a practical application of the methodology; some intermediate mathematical steps will be also referred later.

Proof. Since the assumptions of Theorem 1 are satisfied, every state of the DTMC is positive recurrent and the DTMC is weakly stable. In addition, to prove that the system is strongly stable, we shall show that $\lim_{n \rightarrow \infty} \sup E[\|X_n\|] < \infty$.

Let H_B be the set of values taken by Y_n for which $\|X_n\| \leq B$ (where (7) does not apply). It is easy to prove that H_B is a compact set. Outside this compact set, eq. (7) holds, i.e.

$$E[V(X_{n+1}) - V(X_n) | Y_n] < -\epsilon \|X_n\|$$

Considering all Y_n 's that do not belong to H_B , we obtain

$$E[V(X_{n+1}) - V(X_n) \mid Y_n \notin H_B] < -\epsilon E[||X_n|| \mid Y_n \notin H_B]$$

Instead, for $Y_n \in H_B$, being H_B a compact set,

$$E[V(X_{n+1}) \mid Y_n \in H_B] \leq M < \infty$$

where M is the maximum value taken by $E[V(X_{n+1}) \mid Y_n]$ for Y_n in H_B .

By combining the two previous expressions, we obtain

$$\begin{aligned} E[V(X_{n+1})] &< \\ MP\{Y_n \in H_B\} + P\{Y_n \notin H_B\} \{E[V(X_n) \mid Y_n \notin H_B] - \epsilon E[||X_n|| \mid Y_n \notin H_B]\} &< \\ M + E[V(X_n)] - \epsilon E[||X_n||] + M_0 \end{aligned}$$

M_0 is a constant such that

$$M_0 \geq \{-E[V(X_n) \mid Y_n \in H_B] + \epsilon E[||X_n|| \mid Y_n \in H_B]\}P\{Y_n \in H_B\}$$

Note that M_0 is finite, being H_B a compact set. By summing over all n from 0 to $N_0 - 1$, we obtain

$$E[V(X_{N_0})] < N_0 M + E[V(X_0)] - \epsilon \sum_{n=0}^{N_0-1} E[||X_n||] + N_0 M_0$$

Thus, for any N_0 , we can write

$$\frac{\epsilon}{N_0} \sum_{n=0}^{N_0-1} E[||X_n||] < M + \frac{1}{N_0} E[V(X_0)] - \frac{1}{N_0} E[V(X_{N_0})] + M_0$$

$E[V(X_{N_0})]$ is lower bounded by definition; assume $E[V(X_{N_0})] > K_0$. Hence

$$\frac{\epsilon}{N_0} \sum_{n=0}^{N_0-1} E[||X_n||] < M + \frac{1}{N_0} E[V(X_0)] - \frac{K_0}{N_0} + M_0$$

For $N_0 \rightarrow \infty$, being $E[V(X_0)]$ and K_0 finite, we can write

$$\frac{\epsilon}{N_0} \sum_{n=0}^{N_0-1} E[||X_n||] < M + M_0 \quad (8)$$

Hence $\lim_{N_0 \rightarrow \infty} \frac{1}{N_0} \sum_{n=0}^{N_0-1} E[||X_n||]$ is bounded. Since the DTMC Y_n has positive recurrent states, there exists $\lim_{n \rightarrow \infty} E[||X_n||]$. Furthermore, if the sequence $E[||X_n||]$ is convergent, the sequence $\frac{1}{n} \sum_{i=0}^{n-1} E[||X_i||]$ converges to the same limit (being the Cesaro sum):

$$\lim_{n \rightarrow \infty} E[||X_n||] = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} E[||X_i||]$$

But the right hand side was seen to be bounded; hence, $\lim_{n \rightarrow \infty} E[||X_n||] < \infty$

A class of Lyapunov functions is of particular interest:

Corollary 2. *Given a system of queues as in Theorem 2, then, if there exists a symmetric copositive⁵ matrix $Z \in \mathbb{R}^{N \times N}$, and two positive real numbers $\epsilon \in \mathbb{R}^+$ and $B \in \mathbb{R}^+$, such that, given the function $V(X_n) = X_n Z X_n^T$, $\forall Y_n : \|X_n\| > B$, it holds*

$$E[V(X_{n+1}) - V(X_n) | Y_n] < -\epsilon \|X_n\| \quad (9)$$

then the system of queues is strongly stable. In addition, all the polynomial moments of the queue size distribution are finite.

This is a re-phrasing of the results presented in [18, Sect.IV]. In particular, the identity matrix I is a symmetric positive semidefinite matrix, hence a copositive matrix; thus, it is possible to state the following

Corollary 3. *Given a system of queues as in Theorem 2, if there exist $\epsilon \in \mathbb{R}^+$, $B \in \mathbb{R}^+$ such that $\forall Y_n : \|X_n\| > B$*

$$E[X_{n+1} X_{n+1}^T - X_n X_n^T | Y_n] < -\epsilon \|X_n\| \quad (10)$$

then the system of queues is strongly stable, and all the polynomial moments of the queue size distribution are finite.

2.4 Lyapunov methodology to bound queue sizes and delays

The Lyapunov methodology is also useful to evaluate some bounds on average queue size and average delay of a single IQ switch, as described in [22, 29].

The key observation is that the proof of Theorem 2 provides a first bound on the limit behavior of $E[\|X_n\|]$. Indeed, from (8):

$$\lim_{n \rightarrow \infty} E[\|X_n\|] \leq \frac{1}{\epsilon} (M + M_0) \quad (11)$$

where M is the maximum taken by $E[V(X_{n+1}) | Y_n]$ for $Y_n \in H_B$, where $H_B = \{Y_n, \|X_n\| \leq B\}$, and M_0 is a constant such that

$$M_0 \leq \{-E[V(X_n) | Y_n \in H_B] + \epsilon E[\|X_n\| | Y_n \in H_B]\} P\{Y_n \in H_B\}$$

Unfortunately, this bound is often very loose; thus, tighter bounds can be obtained by selecting special classes of Lyapunov functions.

Considering a system of queues satisfying the assumptions of Theorem 2, since the DTMC describing the evolution of the queues is positive recurrent, if we assume aperiodicity, the DTMC is ergodic. Moreover, since the system is strongly stable:

$$\lim_{n \rightarrow \infty} E[X_{n+1}] = \lim_{n \rightarrow \infty} E[X_n] < \infty$$

⁵ An $N \times N$ matrix Q is copositive if $XQX^T \geq 0 \quad \forall X \in \mathbb{R}^{+N}$

In addition, if the Lyapunov function $V(X_n)$ is a quadratic form, i.e. $V(X_n) = X_n Z X_n^T$, since all the polynomial moments of X_n are finite, it follows that:

$$\lim_{n \rightarrow \infty} E[V(X_{n+1}) - V(X_n)] = 0 \quad (12)$$

An extension to Theorem 2 can be easily obtained by replacing in (7) $-\epsilon \|X_n\|$ with $-\epsilon f(\|X_n\|)$, where $f(\cdot)$ is a continuous function defined on \mathbb{R}^+ . In this case with steps similar to those in the proof of Theorem 2, it is possible to prove that $\lim_{n \rightarrow \infty} E[f(\|X_n\|)] < \infty$.

It is now possible to state the following theorem (from [22]) that provides a stronger and more general bound than (11).

Theorem 3. *Given a system of queues whose evolution is described by a DTMC with state vector $Y_n \in \mathbb{N}^M$, whose state space H is a subset of the Cartesian product of a denumerable state space H_X and a finite state space H_K , and for which all the polynomial moments of the queue sizes distribution are finite, if a lower bounded polynomial function $V(X_n)$, $V : \mathbb{N}^N \rightarrow \mathbb{R}$, can be found, such that*

$$E[V(X_{n+1}) | Y_n] < \infty \quad \forall Y_n$$

and there exist two positive real numbers $\epsilon \in \mathbb{R}^+$ and $B \in \mathbb{R}^+$, such that

$$E[V(X_{n+1}) - V(X_n) | Y_n] \leq -\epsilon f(\|X_n\|) \quad \forall Y_n : \|X_n\| > B \quad (13)$$

being $f(x)$ a continuous function in \mathbb{R}^+ , then

$$\begin{aligned} \lim_{n \rightarrow \infty} E[f(\|X_n\|)] &\leq \\ \lim_{n \rightarrow \infty} E \left[f(\|X_n\|) + \frac{V(X_{n+1}) - V(X_n)}{\epsilon} \mid Y_n \in H_B \right] &P\{Y_n \in H_B\} \end{aligned} \quad (14)$$

2.5 Application to a single queue

Consider a simple discrete-time (or time-slotted) $Geo^{(b)}/D/1$ queue with infinite buffer, where customers arrive in batches of size b at geometrically spaced time intervals, and require a deterministic service time (equal to one time slot). Such a queue can provide a simplified model for either a multiplexer of cell flows, or an output interface of an OQ cell switch, and can serve as an illustrative example of the methodology that we later apply to more complex queuing models of IQ and CIOQ cell switches.

Let x_n be the number of customers in the queue at time slot n ; let a_n be the number of arrivals, and d_n the number of departures, during time slot n . Let $\lambda = E[a_n]$ be the average arrival rate. Observe that both $E[a_n^2]$ and λ are finite.

The equation describing the evolution of this system over time is:

$$x_{n+1} = x_n + a_n - d_n \quad (15)$$

where

$$d_n = \begin{cases} 1 & \text{if } x_n > 0 \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

This simple queuing model corresponds to an irreducible discrete-time Markov chain (DTMC).

Now consider the following linear Lyapunov function: $V(x_n) = x_n$. If we fix $x_n > 0$, then we can write (5) as follows:

$$E[V(x_{n+1}) - V(x_n) \mid x_n > 0] = E[a_n - d_n] = \lambda - 1$$

Hence, thanks to Theorem 1, the queue is weakly stable for $\lambda < 1$.

Now consider the following quadratic Lyapunov function: $V(x_n) = x_n^2$. If we let $x_n \rightarrow \infty$, then we can write (7) as follows:

$$E[V(x_{n+1}) - V(x_n) \mid x_n] = E[2(a_n - d_n)x_n + (a_n - d_n)^2 \mid x_n] = 2(\lambda - 1)x_n + E[(a_n - 1)^2]$$

Now,

$$\lim_{x_n \rightarrow \infty} \frac{E[V(x_{n+1}) - V(x_n) \mid x_n]}{x_n} < -2(1 - \lambda) \quad (17)$$

and Theorem 2 proves the strong stability of the queue in the case $\lambda < 1$.

After evaluating (17), we can apply Theorem 3, using $f(\|X_n\|) = X_n$, to bound the average queue size $E[x]$. Letting $x_n \rightarrow \infty$:

$$\begin{aligned} \lim_{x_n \rightarrow \infty} E[x_n] &\leq \lim_{x_n \rightarrow \infty} E \left[x_n + \frac{-2(1 - \lambda)x_n + (a_n - d_n)^2}{2(1 - \lambda)} \mid x_n \right] \\ &= \lim_{x_n \rightarrow \infty} E \left[\frac{(a_n - d_n)^2}{2(1 - \lambda)} \right] = E \left[\frac{a_n^2 - 2a_n d_n + d_n^2}{2(1 - \lambda)} \right] \\ &= \frac{E[a_n^2] - 2\lambda^2 + \lambda}{2(1 - \lambda)} \end{aligned}$$

being $E[d_n] = \lambda$ because of ergodicity, and $E[d_n] = E[d_n^2]$ because d_n is a binary variable.

Note that the result obtained is the equivalent of the Pollaczek-Khinchin formula [17] for our discrete-time $Geo^{(b)}/D/1$ queue.

From queue size bounds, the derivation of bounds on the average cell delay is easy, thanks to Little's result.

The procedure for the derivation of bounds on the queue size variance is identical, but requires the use of a different functions: $V(x_n) = x_n^3$. We omit the details.

2.6 Final Remark

We notice that the stochastic Lyapunov methodology is a rather simple and powerful tool which has been successfully applied to prove either weak stability or strong stability of several complex systems of queues, such as IQ switches.

The application of stochastic Lyapunov function methodology imposes, however, some rather strong assumptions on the exogenous $\{A_n\}$ arrival process. Even if some extensions to more general cases are possible, usually the sequence of variables $\{X_n\}$, is required to be a DTMC; and thus, the exogenous arrival process at the system, $\{A_n\}$, needs to form a i.i.d. random variable sequence.

More advanced analytical tools, such as *fluid models*, can be applied to prove rate stability under relaxed assumptions on the exogenous arrival process $\{A_n\}$ (fluid models only require $\{A_n\}$ to be a stationary arrival process satisfying the strong law of large numbers).

3 Performance of a single switch

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 $P \times P$ IQ or CIOQ switches). 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 Virtual Output Queueing (VOQ) policy: one separate queue is maintained at each input for each input-output couple. We do not model possible output queues since they never become unstable under admissible traffic patterns. The total number of input queues in each switch is $N = P^2$, which is also equal to the number of customer classes in the general queueing model: $J = N$.

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

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

We consider a synchronous operation, in which the switch configuration can be changed only at slot boundaries. We call *internal time slot* the time necessary to transmit a cell at an input port (or to receive it at an output port). 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 speedup, 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

VOQ 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.

In the following, we will discuss the stability properties for IQ switches and CIOQ switches with speedup 2; in addition, we will show some delay bounds for IQ switches. Before proceeding, we introduce few other mathematical notations.

We adapt the definition of $\|Z\|_{\max L}$ to the case of the single switch.

Definition 6 Given a vector $Z \in \mathbb{R}^N$, $Z = (z^{(k)})$, $k = Pi + j$, $i, j = 0, 1, \dots, P-1$, 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 7 At each time slot, the scheduler of an IQ switch selects for transfer from queues $Q = (q^{(k)})$ a set of cells denoted by vector $D \in \mathbb{N}^N$, $D = (d^{(k)}) \in \{0, 1\}$, $k = Pi + j$, $i, j = 0, 1, \dots, P-1$ 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 8 The traffic pattern loading an (isolated) IQ switch is admissible if and only if $\|E[E_n]\|_{IO} = \|A\|_{IO} < 1$.

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

A traffic is said to be uniform if $A^{(i,j)} = \rho/P$, for $0 \leq i, j < P$, and $0 < \rho < 1$.

Finally we say that a system of queue achieves 100 % throughput when it results strongly stable under any admissible i.i.d arrival process.

3.1 Stability region of pure IQ switches

We say that an IQ switch adopts a Maximum Weight Matching (MWM) scheduling policy if the selection of the switching vector in each slot is implemented according to the following rule:

$$D_n = \arg \max_{D_i \in \mathcal{D}_X} W_n D_i^T \quad (18)$$

where W_n is a vector of weights associated to VOQs, and \mathcal{D}_X denotes the set of all possible $P!$ switching vectors at time n . Note that, for any $D \in \mathcal{D}_X$, $W_n D^T$ represents the weight of matching D .

When the policy maximizes the number of cells to transfer (corresponding to binary case, i.e. $w_n^{(Pi+j)} = 1$ if the corresponding VOQ is not empty, 0 otherwise), the policy computes a maximum *size* matching.

Note also that a matching is said to be *maximal* when no other edge can be added, without violating the switching constraints $\|D\|_{IO} \leq 1$; in general, a maximal matching may be not maximum.

Pure IQ switches (i.e., switches with no speedup) implementing a MWM scheduling algorithm were proved to achieve the same performance in terms of throughput of OQ switches under a wide class of traffic patterns. This fundamental result was first obtained in [25, 26] under i.i.d. arrival processes, applying the stochastic Lyapunov function methodology, and then extended in [12] for more general arrival processes applying fluid models. To obtain maximum throughput, VOQ weights must be proportional to the queue size (e.g., $W_n = X_n$), or to the age of the head-of-the-line cell (Oldest Cell First policy) in the corresponding VOQ [26], or, finally, to the sum of all cells stored in the corresponding input and output ports (Longest Port First policy) [24].

In more recent years the previous results have been generalized in two main directions:

- works [3, 28] provide more general characterization of scheduling algorithms which guarantee 100% throughput in pure IQ architectures;
- works [15, 28, 34] exploit the system memory (i.e., the fact that X_n and X_{n+1} are strongly correlated) to simplify the scheduling algorithms while guaranteeing 100% throughput for IQ architectures.

In the following we report two results; the first, taken from [3], generalizes the result in [26] on MWM optimality; the second, taken from [34], proposes a simple algorithm which exploits the system memory.

Definition 9 Let $F(X)$ be a regular function⁶ $F \in C^1[\mathbb{R}^{+N} \rightarrow \mathbb{R}^{+N}]$. 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:

$$D_n = \arg \max_{D_i \in \mathcal{D}_{X_n}} F(X_n) D_i^T \quad (19)$$

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

Note that when $F(X) = X$, the above policy corresponds to the usual MWM.

The following is the main stability result, which is an extension of [26].

Theorem 4. 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.:

$$\oint_{\Gamma} F(X) d\Gamma(X)^T = 0 \quad (20)$$

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

⁶ C^n denotes the set of continuous functions with continuous i -th derivative, $1 \leq i \leq n$.

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

$$\liminf_{\|X\| \rightarrow \infty} \frac{\|F(X)\|}{\|X\|} \geq s \quad (21)$$

3. all null elements of X remain null:

$$U[X]F(X) = F(X) \quad (22)$$

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 $\mathcal{L}(X)$:

$$\mathcal{L}(X) = \int_{\Gamma_X} F(Y) d\Gamma_X(Y)^T \quad (23)$$

$$\mathcal{L}(0) = 0 \quad (24)$$

where Γ_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 to consider a straight line Γ_X parallel to vector X . Being $X \in \mathbb{R}^{+N}$, both $F(Y)$ and $d\Gamma_X(Y)$ in (23) 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:

$$\begin{aligned} \frac{E[\mathcal{L}(X_{n+1}) | X_n] - \mathcal{L}(X_n)}{\|X_n\|_2} &= O\left(\frac{\nabla \mathcal{L}(X_n)(E[A] - D_n)^T}{\|X_n\|_2}\right) \\ &= O\left(\frac{F(X_n)(E[A] - D_n)^T}{\|X_n\|_2}\right) \end{aligned} \quad (25)$$

We must now show that (25) is smaller than a negative finite constant. By the Birkhoff-von-Neumann theorem [8], 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 (25), 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, $F(X_n)A' \leq \max_{D^* \in \mathcal{D}_{X_n}} F(X_n)D^{*T} = U[X_n]F(X_n)D_n^T$, 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 (21) 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 ϵ' is a positive constant depending on N and $F(X)$.

Coming back to optimal throughput algorithms which exploit the system memory, we present the landmark policy originally proposed in [34]. We notice that several better engineered policies exploiting the system memory have been proposed later [15, 28]; however due to lack of space we refer to the original papers for this class of policies.

The intuition of the approach in [34] is that, if we assume that weights correspond to queue sizes, then correlation exists between the MWM computed in subsequent time slots; indeed, if D_n is the maximum weight matching computed at time n , it can be easily shown that:

$$|X_n D_n^T - X_{n-1} D_{n-1}^T| \leq 2P$$

This correlation can be exploited by keeping memory of the matching used in the previous time slot; if the previous matching was optimal, now it can be considered a good “guess” of the current optimal matching. In addition, the approach exploits randomness to search, at any time, a possible MWM.

This policy, as originally defined in [34], can be formalized as following. Let I_n be a random matching chosen among all $P!$ possible through, for example, a uniform distribution. Now use D_n as the matching, chosen between I_n and D_{n-1} , with the maximum weight:

$$D_n = \arg \max_{D \in \{I_n, D_{n-1}\}} X_n D^T$$

[33] proves that this policy guarantees 100% throughput.

3.2 Delay bounds for maximal weight matching

Stability proved through Lyapunov methodology is asymptotic, since negative drift is shown in the region $\|X_n\| \rightarrow \infty$. Unfortunately, this region has very limited practical relevance, unless some queue sizes or delay bounds are available.

Bounds on the average queue size (and cell delay, as consequence of Little’s law) in a switch implementing MWM with $W_n = X_n$ was obtained in [22] applying a

particular polynomial Lyapunov functions $V(X_n)$. By applying Theorem 3, in [22] it is shown that:

$$E[\|X_n\|_1] \leq \frac{\|A\|_1 - \|A\|_2^2}{1 - \|A\|_{IO}} P \quad (26)$$

In the case of uniform traffic, a bound on the average size of individual input queues can be easily obtained:

$$E[x^{(i)}] \leq \frac{\rho - \rho^2/P}{1 - \rho} \quad (27)$$

where $\rho = \|A\|_{IO}$ represents the port load. Then by Little's theorem the average cell delay is obtained:

$$E[T] = \frac{E[x^{(i)}]}{E[a^{(i)}]} \leq \frac{P - \rho}{1 - \rho} \quad (28)$$

being $E[a^{(i)}] = \rho/P$.

Other stochastic methodologies, as in [5, 27], have been applied to provide delay bounds.

3.3 Stability region of CIOQ with speedup 2

The implementation of optimal schedulers in pure IQ switches can be rather problematic since their algorithmic complexity is rather large (the execution of MWM requires at least $O(P^3)$, see [32]).

However, simpler schedulers may lead to optimal performance if the switch is provided with a moderate speedup $S > 1$. This consideration has encouraged researchers to look for simple and efficient scheduling policies which can be easily implemented in high bandwidth routers. These policies usually compute a maximal size matching, as in the case of WFA [31], iSLIP [23], 2DRR [20], and many others recently proposed.

An important theoretical investigation of these policies has been provided in [12, 21], where it has been proved that CIOQ switches with speedup 2 implementing any maximal Size Matching scheduling (mSM) algorithm achieve 100% throughput.

Consider any queue $q^{(k)}$, $k = Pi + j$, in the VOQ structure, that stores cells at input i directed to output j . Recall that cells stored in $q^{(k)}$ compete for inclusion in the set of non-contending cells with cells stored in each queue $q^{(k')}$, $k' = Pi + l$ with $l \neq j$, and $q^{(k'')}$, $k'' = Ph + j$ with $h \neq i$.

If $q^{(k)}$ is non-empty, the mSM algorithm generates a set of non-contending cells that comprises at least one cell extracted from the *interfering set* $I^{(k)}$

$$I^{(k)} = \bigcup \{q^{(k')} \cup q^{(k'')}\} \quad (29)$$

(exactly one, if the cell is extracted from $q^{(k)}$; possibly two, if one cell is extracted from a $q^{(k')}$, $k' \neq k$, and one from a $q^{(k'')}$, $k'' \neq k$).

Because CIOQ switches have both input and output queues, in the sequel we will use X_n to indicate the state of the input VOQs, while $O_n \in \mathbb{N}^P$ indicates the state vector of output queues.

Theorem 5. *A CIOQ switch with speedup $S \geq 2$ adopting a mSM scheduler is strongly stable under any admissible traffic pattern.*

Stability of mSM scheduling algorithms with $S = 2$ was first proved under a weaker sense (rate stability) in [12] applying the fluid models methodology, and then strengthened in [21, 22] applying the stochastic Lyapunov function methodology. The proof we report is taken from [22].

Proof. Consider as Lyapunov function $V(X_n) = X_n Q X_n^T$, where $Q \in \mathbb{R}^N \times \mathbb{R}^N$ is such that

$$q_{ij} = \begin{cases} 1 & \text{if } j = (i + lP) \bmod N, l = 0, \dots, P-1 \\ 1 & \text{if } P\lfloor i/P \rfloor \leq j < P\lfloor i/P \rfloor + P \\ 0 & \text{otherwise} \end{cases}$$

i.e., $X_n Q = \sum_{l \in I^{(n)}} x_n^{(l)}$ is the number of cells stored in interfering queues. Figure 2 reports a sketch of the Q matrix structure.

The figure shows the schematic structure of the matrix Q . It is represented as a sum of two matrices. The first matrix has a block structure where the diagonal elements are 1, and there are diagonal blocks of size P at intervals of P . The second matrix has a similar structure but with vertical bars representing the blocks.

Fig. 2. Schematic block for the Q operator used as Lyapunov function.

We need to prove that inequality (7) holds to prove the system stability. Thus, recalling that $X_{n+1} = X_n + A_n - \mathcal{D}_n$, we have

$$\begin{aligned} E[V(X_{n+1}) - V(X_n) | X_n] &= \\ &= E[2X_n Q (A_n - \mathcal{D}_n)^T + (A_n - \mathcal{D}_n) Q (A_n - \mathcal{D}_n)^T | X_n] \end{aligned}$$

being Q a symmetric matrix. For $\|X_n\|_1 \rightarrow \infty$, it holds

$$\begin{aligned} E[2X_n Q (A_n - \mathcal{D}_n)^T + (A_n - \mathcal{D}_n) Q (A_n - \mathcal{D}_n)^T | X_n] &= \\ &= 2E[X_n Q A_n^T | X_n] - 2E[X_n Q \mathcal{D}_n^T | X_n] + o(\|X_n\|_1) \end{aligned}$$

where $\lim_{\|X_n\|_1 \rightarrow \infty} \frac{o(\|X_n\|_1)}{\|X_n\|_1} = 0$. But $E[A_n]Q \leq \|A\|_{\max L} \mathbb{I}$ and

$$E[X_n Q A_n^T | X_n] = E[A_n] Q X_n^T \leq \|A\|_{\max L} \|X_n\|_1 \quad (30)$$

At the same time, for the definition of the mSM algorithm, which selects \mathcal{D}_n comprising at least one cell from interfering set $I^{(k)}$, and given the speedup $S \geq 2$,

$$\begin{aligned} (E[D_n]Q)^{(k)} X_n^{(k)} &\geq 2x_n^{(k)} - 1P\{x_n^{(k)} = 1 \wedge d_n^{(k)} = 1\} \\ &\geq 2x_n^{(k)} - P\{d_n^{(k)} = 1\} = 2x_n^{(k)} - E[d_n^{(k)}] = 2x_n^{(k)} - \lambda^{(k)} \end{aligned}$$

Thus

$$E[X_n Q \mathcal{D}_n^T | X_n] \geq 2\|X_n\|_1 - \|A\|_1 \quad (31)$$

Thus, there exist $B > 0$ and $\epsilon > 0$ such that:

$$\begin{aligned} E[V(X_{n+1}) - V(X_n) | X_n] &= 2E[X_n Q A_n^T | X_n] - 2E[X_n Q \mathcal{D}_n^T | X_n] + o(\|X_n\|_1) \\ &\leq -\epsilon \|X_n\|_1 \quad \forall X_n : \|X_n\|_1 > B \end{aligned}$$

Finally, we emphasize that several other scheduling policies which do not fall in the class of mSM have been shown to achieve optimal throughput in switches with speedup $S \geq 2$ [21]. As an example, any scheduling policy according to which the selected departure vector D_n is such that:

$$X_n D_n^T > \frac{1}{2} \max_{D \in \mathcal{D}_X} X_n D^T$$

was proved to achieve optimal throughput in switches with speedup $S \geq 2$ [21].

3.4 Scheduling variable size packets

All the results shown so far through the Lyapunov methodology assume that fixed size cells are switched across the switching fabric. To apply these results in the context of an IP router, the hardware architecture must include some modules which, at the inputs, chop the variable-size packets into fixed-size cells and, at the outputs, reassemble the cells. Even if this architecture is common in practice, it requires additional complexity to handle the conversion between packets and cells; hence, other architectures have been designed to switch natively variable-size packets. Fortunately, some theoretical results have been produced in recent years on the achievable throughput of IQ switches handling variable size packets [1, 14].

Variable length packets are modelled in this context as trains of fixed size cells which have to be transferred to output ports through synchronous fabrics in contiguous time slots. In paper [1], applying the stochastic Lyapunov function methodology, it has been proved that a pure IQ switch with no speedup, implementing a variant of

the MWM algorithm allowing the transfer of cells originated by the same packet in contiguous time slots, still achieves 100% throughput.

More precisely, denote with \mathcal{S}_n the set of VOQs from which the transfer of a packet is in act at time slot n .

Theorem 6. *An IQ switch with no speedup is strongly stable under any admissible traffic pattern if at each time slot the departure vector is selected according to a MWM scheduling algorithm:*

$$D_n = \arg \max_{D_i \in \mathcal{D}_x} W_n D_i^T \quad (32)$$

in which VOQ weights are:

$$w_n^{(k)} = \begin{cases} x_n^{(k)} & \text{if } k \notin \mathcal{S}_n \\ \infty & \text{if } k \in \mathcal{S}_n \end{cases}$$

This result was extended in [14] under a wider class of arrival processes by applying the fluid model methodology.

4 Networks of IQ switches

Consider the case of a network interconnecting many IQ switches, each of them running a MWM scheduling algorithm, which guarantees to achieve 100% throughput when each switch is studied in isolation. It was shown in [6] that a specific network of IQ switches implementing a MWM scheduling policy can exhibit an unstable behavior even when none of the switches are overloaded. This 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 [6] 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 in the network, thus relying on an exact knowledge of the traffic pattern at each switch, an approach not feasible in practice. In addition, the result proved in [6] cannot be easily extended to more general traffic patterns in which flows are not leaky bucket compliant.

As an example of counterintuitive behavior, consider the network of eight IQ switches depicted in Fig. 3, 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 adjacent IQ switches (all pairs are alike) is traversed by a locally originated flow, a locally terminating flow, and an in-transit flow. We run simulation experiments in which 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; hence, the traffic is admissible. In-transit and terminating flows are given weight 10 times larger than locally originating flows. Fig. 4 shows that queue sizes take a divergent

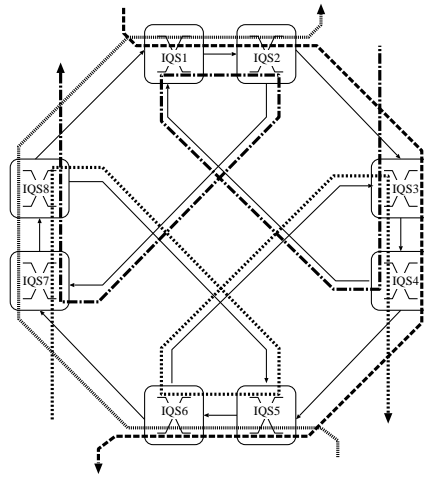


Fig. 3. The network of IQ switches considered in our simulation.

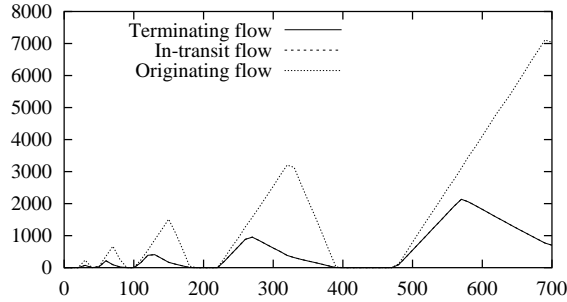


Fig. 4. Queue sizes versus time in slots for the three flows at IQS1, when a local MWM is computed at each switch.

oscillating behavior when a local MWM scheduling is adopted.

We show in the sequel how to modify the MWM to guarantee the maximum throughput in a network.

4.1 Theoretical performance

We consider a network of K IQ 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 adopts VOQ at inputs. Thus there are P_k^2 different VOQs at switch k .

Thus, the network of switches can be modelled as a system Q containing $N = \sum_k P_k^2$ virtual queues. We restrict our study to the case $P_k = P \forall k$, so that $N = P^2 K$. Let $S(q)$ be the function that returns the switch on which VOQ q is located; let

$I(q)$ be the function that returns the index of the input card at switch $S(q)$ on which the VOQ is located; let $O(q)$ be the function that returns the index of the output card at switch $S(q)$ to which VOQ cells are directed.

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

Definition 10 Given a vector $Z \in \mathbb{R}^N$, $Z = \{z^{(n)}, n = P^2k + Pi + j, 0 \leq k < K, i, j = 0, 1, \dots, P - 1\}$, the norm $\|Z\|_{IO}$ is defined as:

$$\|Z\|_{IO} = \max_{\substack{k = 0, \dots, K - 1 \\ i = 0, \dots, P - 1}} \left\{ \sum_{n \in S^{-1}(k) \cap O^{-1}(i)} |z^{(n)}|, \sum_{n \in S^{-1}(k) \cap I^{-1}(i)} |z^{(n)}| \right\} \quad (33)$$

At each time slot, a set of non contending cells departs from the VOQs of each switch. More formally, we say that, at each time slot, the departure vector $D \in \{0, 1\}^N$ must satisfy the condition:

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

The $N \times N$ matrix $R = [r^{(k,l)}]$ is the routing matrix: element $r^{(k,l)} = 1$ for cells departing from VOQ k and entering VOQ l .

Definition 11 The traffic pattern loading a network of IQ switches is admissible if and only if

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

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

The following results are taken from [3].

Theorem 7. An open network of IQ 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 (21) and (22);
- $F(\alpha X) = \alpha F(X)$ for all scalars α .

Similarly to the case of a single switch in isolation, for a network of switches it is possible to extend the result to more general functions $F(X)$ under any admissible i.i.d traffic pattern (i.e., under a smaller class of traffic patterns with respect to the assumptions of Theorem 7), by directly applying the Lyapunov function methodology to equations describing the stochastic evolution of the system.

Theorem 8. An open network of IQ switches implementing the $F(X)$ -max-scalar policy is strongly stable under each i.i.d. admissible traffic pattern, if:

- $G(X) = F(X)[(I - R)^{-1}]^T$ defines a conservative field;
- $F(X)$ satisfies conditions (21) and (22).

As a consequence, it can be shown the following result, corresponding to the policy proposed in [33].

Theorem 9. *An open network of IQ switches implementing the $F(X)$ -max-scalar policy, with $F(X) = X(I - R)^{-1}$ is rate stable under each admissible traffic pattern such that the sequences of arrivals at VOQs satisfy the strong law of large numbers.*

The previous stable policy corresponds to a local MWM matching where the weight $w(q)$ associated to queue q of size $x(q)$ is given by:

$$w(q) = \max\{0, x(q) - x(d[q])\}$$

when $d[q]$ is the downstream queue where cells from queue q are routed after being served⁷.

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 neighbor switches. Some other possible solutions have been studied in [2].

5 Conclusions

In this chapter we have shown how the stochastic Lyapunov function methodology was employed during the last years as a powerful and versatile analytical tool to assess the performance of input queued switches. Thanks to this approach, researchers have been able to study throughput and/or delay performance, referring not only to isolated switches, but also to networks of switches.

The brief discussion of the results reported in this chapter hides the main difficulty in employing this methodology; indeed, the design of a Lyapunov function, specific for the system under study and suited to prove its stability properties, requires often a creative effort, which is the most difficult step in the theoretical investigation. We strongly recommend the interested readers to refer to all the original papers and proofs, to acquire a good “sensitivity” in the identification of the most suited Lyapunov functions for the system under study.

References

1. Ajmone Marsan M, Bianco A, Giaccone P, Leonardi E, Neri F (2002) Packet-mode scheduling in input-queued cell-based switches. In: IEEE/ACM Transactions on Networking 10:666–678

⁷ For a proper definition of $w(q)$, it is defined $x(d[q]) = 0$ when q is an egress queue of the network.

2. Ajmone Marsan M, Giaccone P, Leonardi E, Neri F (2003) On the stability of local scheduling policies in networks of packet switches with input queues. In: IEEE JSAC 21:642-655
3. Ajmone Marsan M, Leonardi E, Mellia M, Neri F (2005) On the Stability of Isolated and Interconnected Input-Queueing Switches Under Multiclass Traffic. In: IEEE Transactions on Information Theory 45:1167-1174
4. Anderson T, Owicki S, Saxe J, Thacker C (1993) High Speed Switch scheduling for local area networks. In: ACM Transactions on Computer Systems 319-352
5. Andrews M, Vojnovic M (2003) Scheduling reserved traffic in input-queued switches: new delay bounds via probabilistic techniques. In: IEEE JSAC 21:595-605
6. Andrews M, Zhang L (2001) Achieving stability in networks of input-queued switches. In: Proc. of IEEE INFOCOM 2001 1673-1679
7. Baskett F, Chandy KM, Muntz RR, Palacios F (1975) Open, closed and mixed networks with different classes of customers. In: Journal of the ACM 22:248-260
8. Cheng-Shang C, Wen-Jyh C, Hsiang-Yi H (2000) Birkhoff-von Neumann input buffered crossbar switches. In: Proc. of IEEE INFOCOM 2000
9. Chuang ST, Goel A, McKeown N, Prabhakar B (1999) Matching output queuing with combined input and output queuing. In: IEEE JSAC 17:1030-1039
10. Dai JG (1999) Stability of Fluid and Stochastic Processing Networks. In: Miscellanea Publication n.9, Center for Mathematical Physics and Stochastic, Denmark (<http://www.maphysto.dk>)
11. Dai JG (1995) On Positive Harris Recurrence of Multiclass Queueing Networks: a Unified Approach Via Fluid Limit Models. In: Annals of Applied Probability 5:49-77
12. Dai JG, Prabhakar B (2000) The throughput of data switches with and without speedup. In: Proc. of IEEE INFOCOM 2000 556-564
13. Fayolle G (1989) On random walks arising in queueing systems: ergodicity and transience via quadratic forms as Lyapunov functions – Part I. In: Queueing Systems 5:167-184
14. Y.Ganjali Y, A.Keshavarzian A, D.Shah D (2005) Cell Switching Versus Packet Switching in Input-Queued Switches. In: IEEE/ACM Transactions on Networking 13:782-789
15. Giaccone P, Prabhakar B, Shah D (2003) Randomized scheduling algorithms for high-aggregate bandwidth switches. In: IEEE JSAC 21:546-559
16. Karol M, Hluchyj M, Morgan S (1987) Input versus output queuing on a space division switch, IEEE Transactions on Communications 35:1347-1356
17. Kleinrock L (1975) Queueing Systems – Vol. I: Theory, J. Wiley
18. Kumar PR, Meyn SP (1995) Stability of queueing networks and scheduling policies. In: IEEE Transactions on Automatic Control 40:251-260
19. Kushner HJ (1967) Stochastic Stability and Control, Academic Press
20. LaMaire RO, Serpanos DN (1994) Two dimensional round-robin schedulers for packet switches with multiple input queues. In: IEEE/ACM Transactions on Networking 2:471-482
21. Leonardi E, Mellia M, Neri F, Ajmone Marsan M (2001) On the stability of Input-Queued Switches with Speed-up. In: IEEE/ACM Transactions on Networking 9:104-118
22. Leonardi E, Mellia M, Neri F, Ajmone Marsan M (2003) Bounds on Delays and Queue Lengths in Input-Queued Cell Switches. In: Journal of the ACM 50:520-550
23. McKeown N (1999) The iSLIP scheduling algorithm for input-queued switches. In: IEEE Transactions on Networking 7:188-201
24. McKeown N, Mekkittikul A (1998) A practical scheduling algorithm to achieve 100% throughput in input-queued switches. In: Proc. of IEEE INFOCOM'98 792-799

25. McKeown N, Anantharam V, Walrand J (1996) Achieving 100% Throughput in an Input-Queued Switch. In: Proc. of IEEE INFOCOM'96 296-302
26. McKeown N, Mekkittikul A, Anantharam V, Walrand J (1999) Achieving 100% throughput in an input-queued switch. In: IEEE Transactions on Communications 47:1260-1267
27. Neely NJ, Modiano E (2004) Logarithmic delay for NxN packet switches. In: Proc. of IEEE HPSR 2004 3-9
28. Ross K, Bambos N (2004) Local search scheduling algorithms for maximal throughput in packet switches. In: Proc. of IEEE INFOCOM 2004 2:1158-1169
29. Shah D, Kopikare M (2002) Delay Bounds for Approximate Maximum Weight Matching Algorithms for Input Queued Switches. In: Proc. of IEEE INFOCOM 2002
30. Tamir Y, Frazier G (1988) High performance multi-queue buffers for VLSI communication switches. In: Proc. of 15th Annual Symposium on Computer Architecture 343-354
31. Tamir Y, Chi HC, Symmetric crossbar arbiters for VLSI communication switches (1993). In: IEEE Transactions on Parallel and Distributed Systems 4:13-27
32. Tarjan RE (1983) Data structures and network algorithms, Society for Industrial and Applied Mathematics, Pennsylvania
33. Tassiulas L, Ephremides A (1992) Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. In: IEEE Transactions on Automatic Control 37:1936-1948
34. Tassiulas L (1998) Linear complexity algorithms for maximum throughput in radio networks and input queued switches. In: Proc. of IEEE INFOCOM'98 553-559