Stability of a Frame-Based Oldest-Cell-First Maximal Weight Matching Algorithm

Xike Li, *Student Member, IEEE*, and Itamar Elhanany, *Senior Member, IEEE*

***Abstract*— Input-queued cell switches employing the oldest-cell-first (OCF) policy have been shown to yield low mean delay characteristics. Moreover, it has been proven that OCF is stable for admissible traffic conditions when executed with a scheduling speedup of 2. However, as link speeds increase, the computational complexity of these algorithms limits their applicability in high port-density switches and routers. To address the scalability issue, we describe a Frame-Based Maximal Weight Matching (FMWM) algorithm, employing OCF as queue weights, in which a new scheduling decision is issued once every several cell times. Between scheduling decisions, the configuration of the crossbar switch remains unchanged. We further extend the analysis to address the case of multiple classes of service, and prove that the algorithm is stable with an internal buffer transfer speedup of 2, thereby significantly relaxing the timing constraints imposed on the scheduling process.**

***Index Terms*— Packet scheduling algorithms, stability analysis, switching architectures.**

I. Introduction

INPUT-QUEUED cell switching architectures are commonly utilized in Internet routers as they offer pragmatic scalability while requiring moderate memory bandwidth. In such architectures, arriving cells (i.e. fixed-size packets) are buffered at the ingress ports before traversing a crossbar switch en route to their destination (egress) ports. A common technique for overcoming potential blocking and congestion at the input ports is virtual output queueing (VOQ) [11]. In VOQ, a separate queue is maintained at the ingress port for each of the N output destinations. A scheduler is responsible for determining a matching configuration between inputs and outputs, whereby at any given time at most one input is matched to one output, and vice versa.

A switch with a speedup of 1 is said to allow at most one cell from each input to traverse the crossbar during a single time (cell) slot. If a switch has a speedup of s, where $s \in \{1, ..., N\}$, it is said to issue s scheduling decisions, and correspondingly s transmissions of cells, from input queues to output ports during a single time slot. Observably, when $s > 1$, buffering is required at the output ports. Such architectures are commonly referred to as combined input-and-output-queued (CIOQ)[2] switches. Many scheduling algorithms have been proposed for CIOQ switches in recent years, with a common goal of offering scalability together with high-performance. In the context of the latter, a fundamental requirement from any scheduling algorithm is stability. Stated coarsely, a switch is said to be stable if all its queues are bounded and, hence, never backlog indefinitely. Once a switch has been proven to be stable, its performance can be evaluated by means of simulations with reasonable confidence.

It has been shown that for a broad class of traffic arrival patterns, all *maximal matching algorithms* yield a stable switch of any size with a speedup of 2 [2], [6], [9]. This stability property holds while delivering throughput of up to 100%. The majority of the work published addresses scenarios in which either the queues size (longest-queue first - LQF) [8] or the age of the head-of-line cell (oldest-cell-first - OCF) [7] is used as a metric reflecting on the queues' priority/urgency. It has been shown that when OCF is employed, the variance of the average packet delays is kept to a minimum [9], rendering it more attractive from a performance perspective.

A subset of maximal matching algorithms is maximal weight matching (MWM) algorithms, in which greedy convergence to a maximal aggregate matching weight is obtained. As with all maximal matching algorithms, MWM requires $O(N)$ iterations to converge, where iteration refers to a single input-output matching. In light of the latter, it is interesting to note that the increase in link rates (e.g. 40 Gbps) has directly resulted in decreased cell durations to a point where cell-by-cell switching is no longer feasible. This is a key hurdle in building high port density next-generation switches and routers.

In this paper, we propose to employ a frame-based *maximal* weight matching (FMWM) algorithm, in which scheduling decisions are issued in accordance with the MWM algorithm, however they are kept unchanged for a duration of k consecutive time slots. By reconfiguring the crossbar switch once every several time slots we significantly relax the timing constraints imposed on the scheduling algorithm [4], thus allowing the system to scale. The age of the oldest cell in the queue is used by the scheduler as the queue's weight, thereby reflecting on urgency of service. Moreover, we extend our analysis to address the scenario of multiple classes of service in which a weighted priority scheme is applied. We further prove that stability is guaranteed for a large class of frame-based QoS policies. While the notion of frame-based switching has been investigated in the literature [1][5], the latter have focused mainly on heuristic simulation studies and basic scheduling schemes.

The rest of the paper is structured as follows. Section II is dedicated to the stability proof of the OCF-based FMWM (FMWM/OCF) scheduling algorithm for the case of a single class of service. In Section III, the analysis is extended

Paper approved by G. S. Kuo, the Editor for Communication Architectures of the IEEE Communications Society. Manuscript received July 30, 2005; revised October 16, 2006 and February 12, 2007. This work has been partially supported by the Department of Energy (DOE) under research grant DE-FG02-04ER25607.

The authors are with the Department of Electrical Engineering and Computer Science at The University of Tennessee, Knoxville, TN 37996 USA (e-mail: {xli6, itamar}@utk.edu).

Digital Object Identifier 10.1109/TCOMM.2008.050189.

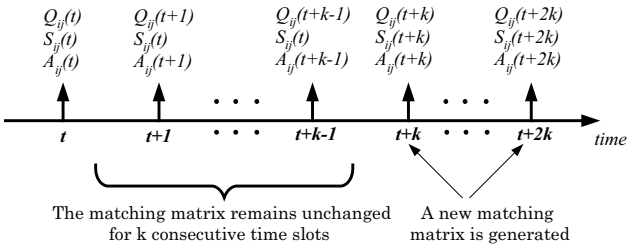

Fig. 1. Buffer dynamics under the FMWM scheduling algorithm.

to address multiple class of service. Section IV discusses simulation results, while in Section V the conclusions are drawn.

II. FMWM/OCF with a Single Class of Service

Consider a CIOQ switch employing VOQ with N ports. Let $Q_{ij}(t)$ denote the virtual output queue size at input i buffering cells destined to output j at time t. We define the random process, $A_{ij}(t) \in \{0,1\}$, with a mean rate of cell arrivals from input i to output j denoted by $E[A_{ij}(t)] = \lambda_{ij} \leq 1$. We consider the simple FMWM which consists on an iterative process whereby in each iteration the largest weight is found and a match is recorded between its associated input-output pair. Each time a match is generated, the respective input and output are removed from contending during consecutive iterations. Consequently, all weights associated either with input i or output j are removed such that they are not considered in subsequent iterations. Assuming the weight matrix is not completely null, the number of iterations ranges from 1 to N.

The configuration of the crossbar, which is the outcome of the algorithm, can be represented by the permutation matrix. $S(t) = \{S_{ij}(t)\}$, where $S_{ij}(t) = 1$ if input i is matched to output j at time t, otherwise $S_{ij}(t) = 0$. Without loss of generality, let us assume that at time t we have explicit knowledge of age of the head-of-line (HoL) cells. We thus assign a weight value equal to the age of the HoL packet, to each virtual output queue, based on which the scheduler establishes the matchings for k consecutive time slots. As a result, at time $t+k$, we have a new matching/scheduling matrix $S_{ij}(t+k)$. As depicted in Fig. 1, the matching matrix remains unchanged during the following k consecutive time slots. We begin with several basic definitions used throughout the rest of the paper.

Definition 1: An arrival process is said to be strictly admissible if

$$\sum_{i=1}^{N} \lambda_{ij} \leq 1, \ \sum_{j=1}^{N} \lambda_{ij} \leq 1. \tag{1}$$

Definition 2: Let the *queue length vector* at time t be defined as

$$Q(t) = \left[Q_{11}(t), \ \cdots, \ Q_{1N}(t), \ \cdots, \ Q_{NN}(t)\right]^T, \tag{2}$$

Definition 3: Let the *waiting time vector of the HoL cells* at time t be

$$\Omega(t) = \left[T_{11}(t), \ \cdots, \ T_{1N}(t), \ \cdots, \ T_{NN}(t)\right]^T \tag{3}$$

where $T_{ij}(t)$ is the waiting time of the HoL cell in queue (i,j) at time t.

Definition 4: Let $\tau_{ij}^{(m)}(t)$ denote the inter-arrival time between two consecutive cells, m and $m+1$, both of which are stored in queue (i,j), and correspondingly, let $\tau_{ij}(t) = \max\{\tau_{ij}^{(m)}(t),\ m = 1,2,...,Q_{ij}(t)\}$.

Definition 5: The *transfer speedup*, $\eta \geq 1$, shall be referred to as the maximal number of cells that each input queue can transfer to the output ports during a single time slot.

It is important to emphasize that while in the literature the notion of speedup entails the requirement for multiple *scheduling* decisions to be issued in each time slot, the concept of transfer speedup is computationally much more relaxed as it does not require multiple scheduling decisions. Next we will give theoretical stability analysis based on Lyapunov methodology. The theoretical foundation can be found in [3]

Theorem 1: A CIOQ switch employing the FMWM/OCF scheduling algorithm with a transfer speedup of 2 is stable under admissible i.i.d. traffic for any frame size k.

Proof: Our proof will focus on deriving a speedup sufficiency condition under which the system will be stable. The buffer dynamics under the FMWM algorithm dictate that for $Q_{ij}(t) > \eta k$

$$T_{ij}(t+k) = T_{ij}(t) + k - S_{ij}(t) \sum_{m=1}^{\eta k} \tau_{ij}^{(m)}(t) \tag{4}$$

while for $Q_{ij}(t) \leq \eta k$

$$\begin{aligned} & T_{ij}(t+k) \leq T_{ij}(t) + k \leq Q_{ij}(t) \cdot \tau_{ij}(t) + k \quad (5) \\ \leq \ & \eta k \tau_{ij}(t) + k \end{aligned}$$

Next, we construct a discrete-time quadratic Lyapunov function, $L(t)$, such that $L(t) = \sum_{ij} \lambda_{ij} T_{ij}^2(t)$. Consequently, we have

$$L(t+k) - L(t) = \sum_{ij} \lambda_{ij} T_{ij}^2(t+k) - \sum_{ij} \lambda_{ij} T_{ij}^2(t)$$

By partitioning the above into the case of $Q_{ij}(t) \leq \eta k$ and $Q_{ij}(t) > \eta k$, we deduct that if $Q_{ij}(t) > \eta k$, given equation 4, we have

$$\begin{aligned} & T_{ij}^2(t+k) - T_{ij}^2(t) \quad (6) \\ = \ & \left[k - S_{ij}(t) \sum_{m=1}^{\eta k} \tau_{ij}^{(m)}(t)\right]^2 \\ + \ & 2\left[k - S_{ij}(t) \sum_{m=1}^{\eta k} \tau_{ij}^{(m)}(t)\right] T_{ij}(t) \end{aligned}$$

hence,

$$\begin{aligned} & T_{ij}^2(t+k) - T_{ij}^2(t) \quad (7) \\ \leq \ & k^2 + 2\left[k - S_{ij}(t) \sum_{m=1}^{\eta k} \tau_{ij}^{(m)}(t)\right] T_{ij}(t) \end{aligned}$$

and if $Q_{ij}(t) \leq \eta k$

$$\begin{aligned} & T_{ij}^2(t+k) - T_{ij}^2(t) \leq k^2 + 2kT_{ij}(t) \quad (8) \\ \leq \ & k^2 + 2\eta k^2 \tau_{ij}(t) \end{aligned}$$

Therefore, by applying equations (7) and (8), we observe that the drift in the Lyapunov function is

$$\begin{aligned} & E\left[L(t+k)-L(t)|\Omega(t)\right] \qquad (9)\\ \leq\ & \sum_{ij}\lambda_{ij}\left(k^2+2\left[k-S_{ij}(t)\sum_{m=1}^{\eta k}E\left[\tau_{ij}^{(m)}(t)\right]\right]T_{ij}(t)\right)\\ & +\sum_{ij}\lambda_{ij}E\left[k^2+2\eta k^2\tau_{ij}(t)\right]\\ \leq\ & 2\sum_{ij}\lambda_{ij}\left[k-\frac{\eta k S_{ij}(t)}{\lambda_{ij}}\right]T_{ij}(t)+C \end{aligned}$$

where $C=2\sum\limits_{ij}\lambda_{ij}k^2+2\sum\limits_{ij}\eta k^2$. Hence, we conclude that

$$\begin{aligned} & E\left[L(t+k)-L(t)|\Omega(t)\right] \qquad (10)\\ \leq\ & 2\sum_{ij}\lambda_{ij}\left[k-\frac{\eta k S_{ij}(t)}{\lambda_{ij}}\right]T_{ij}(t)+C\\ \leq\ & 2k\left(\langle\Lambda,\Omega(t)\rangle-\eta\left\langle S,\Omega(t)\right\rangle\right)+C \end{aligned}$$

where $\Lambda=\|\lambda_{ij}\|$ denotes the admissible arrival rate matrix, which is doubly stochastic, and the operator $<,>$ is the dot product.

In order to prove that the algorithm yields a stable queueing system, we would like to show that beyond a given threshold of maximal weight there is a negative drift in the state of the system, as reflected by the Lyapunov function. Mathematically speaking, from inequality (10), an appropriate value for η, such that $\langle\Lambda,\Omega(t)\rangle < \eta\langle S(t),\Omega(t)\rangle$, guarantees that the algorithm is stable.

Without loss of generality, assume that following a round of matching, VOQ_{sl}, where $s\epsilon[1,...,N]$ and $l\epsilon[1,...N]$ is selected. It then follows that all of the elements in row s and column l of the weight matrix are removed from future contention. By decomposing $<\Lambda,\Omega>\leq\eta<S,\Omega>$ into each round, we have $\sum_{j=1}^{N}\lambda_{sj}+\sum_{i=1}^{N}\lambda_{il}\leq\eta$. On the other hand, for any admissible traffic pattern, we know that $\sum_{j=1}^{N}\lambda_{sj}\leq 1$ and $\sum_{i=1}^{N}\lambda_{il}\leq 1$, from which we deduct that $\sum_{j=1}^{N}\lambda_{sj}+\sum_{i=1}^{N}\lambda_{il}\leq 2$. Hence, $\eta=2$ is sufficient to guarantee stability. ■

III. FMWM/OCF WITH MULTIPLE CLASSES OF SERVICE

Next, we extend the analysis to show that when multiple classes of service are employed at each input port, the same stability property holds. The underlying assumption is that the notion of virtual output queueing is expanded such that there are now a set of L queues residing in input i destined to output j, where by each of the L queues has a different priority level associated with it. The priority is reflected by a weighted scheme such that the age of the oldest cell in the queue is multiplied by a per-class coefficient to yield the weight used by the scheduler to issue the matching decisions. To clarify this point, we shall refer to the following definitions:

Definition 6: With weighted priorities, an arrival process is said to be admissible iff

$$\sum_{il}\lambda_{ijl}\leq 1,\quad \sum_{jl}\lambda_{ijl}\leq 1 \qquad (11)$$

where λ_{ijl} denotes the normalized offered load from input i to output j, in class $l\in[1,L]$.

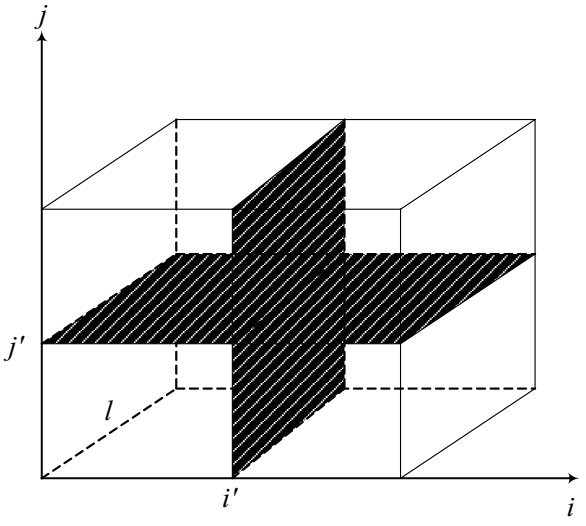

Fig. 2. Upon generating an input-output match, two plains are removed from the weight tensor, thereby reducing the set of contending nodes in subsequent iterations of the algorithm.

Definition 7: Let the scheduling *weight vector* at time t be defined as

$$\Omega(t)=\left[W_{111},\ \cdots,\ W_{1N1},\ \cdots,\ W_{NNL}\right]^T \qquad (12)$$

where $W_{ijl}(t)=C_lT_{ijl}(t)$, $i,j=[1,N]$, $l=[1,L]$, and C_l is the (positive) weight coefficient of class l, such that $\sum C_l=1$.

Definition 8: Let $S(t)=\{S_{ijl}(t)\}$ denote the permutation matrix, where $S_{ijl}(t)=1$ if class l in input i is matched to output j at time t, otherwise $S_{ijl}(t)=0$. Note that the maximal number of possible matches remains N (i.e. $\sum_{i,j,l}S_{ijl}\leq N$), since at most N inputs can be matched to N outputs.

Definition 9: Let $\tau_{ijl}^{(m)}(t)$ denote the inter-arrival time between two consecutive cells, m and $m+1$, both of which are stored in VOQ $Q_{ijl}(t)$, and let $\tau_{ijl}(t)=\max\{\tau_{ijl}^{(m)}(t),\ m=1,2,...,Q_{ijl}(t)\}$.

The configuration of the crossbar, which is the outcome of the FMWM algorithm, can be represented by the permutation tensor of order three, as illustrated in Fig. 2. Without loss of generality, let us assume that at time t we have explicit knowledge of $W_{ijl}(t)$. The FMWM algorithm guarantees that $S_{ijl}(t)=1$ always points to the largest value on row i, column j and class of service l, respectively. Following each iteration, the two plains which contained the largest weight are removed, as illustrated in Fig. 2 (in this example plains $\{i',j,l\}$ and $\{i,j',l\}$ are removed). This leads to the following theorem:

Theorem 2: A CIOQ switch employing the FMWM/OCF scheduling algorithm with a transfer speedup of 2 and multiple classes of service, realized using positive per-class weight coefficients, is stable under admissible i.i.d. traffic for any frame size k.

Proof: Similarly to (4), when $Q_{ijl}(t)>\eta k$ we may write

$$T_{ijl}(t+k)=T_{ijl}(t)+k-S_{ijl}(t)\sum_{m=1}^{\eta k}\tau_{ijl}^{(m)}(t) \qquad (13)$$

and if $Q_{ijl}(t) \leq \eta k$

$$\begin{aligned} T_{ijl}(t+k) &\leq T_{ijl}(t) + k \leq Q_{ijl}(t) \cdot \tau_{ijl}(t) + k \\ &\leq \eta k \tau_{ijl}(t) + k \end{aligned} \quad (14)$$

Applying (13) and (14) with weighted per-classes OCF, the following holds for $Q_{ijl}(t) > \eta k$

$$W_{ijl}(t+k) = W_{ijl}(t) + C_l k - C_l S_{ijl}(t) \sum_{m=1}^{\eta k} \tau_{ijl}^{(m)}(t) \quad (15)$$

otherwise,

$$\begin{aligned} & W_{ijl}(t+k) \leq W_{ijl}(t) + C_l k \\ \leq\ & C_l Q_{ijl}(t) \cdot \tau_{ijl}(t) + C_l k \\ \leq\ & C_l \eta k \tau_{ijl}(t) + C_l k \end{aligned} \quad (16)$$

Next, we construct a discrete-time quadratic Lyapunov function, $L(t)$, in the form $L(t) = \sum_{ijl} \frac{\lambda_{ijl}}{C_l} W_{ijl}^2(t)$. Consequently, we obtain

$$L(t+k) - L(t) = \sum_{ijl} \frac{\lambda_{ijl}}{C_l} \left[W_{ijl}^2(t+k) - W_{ijl}^2(t) \right] \quad (17)$$

From (15) and (16), we have the following inequality

$$\begin{aligned} & L(t+k) - L(t) \\ \leq\ & \sum_{ijl} \frac{\lambda_{ijl}}{C_l} \left[C_l k - C_l S_{ijl}(t) \sum_{m=1}^{\eta k} \tau_{ijl}^{(m)}(t) \right]^2 + \\ & \sum_{ijl} 2 \frac{\lambda_{ijl}}{C_l} \left[C_l k - C_l S_{ijl}(t) \sum_{m=1}^{\eta k} \tau_{ijl}^{(m)}(t) \right] W_{ijl}(t) + \\ & \sum_{ijl} \frac{\lambda_{ijl}}{C_l} C_l^2 \left[k^2 + 2\eta k^2 \tau_{ijl}(t) \right] \end{aligned} \quad (18)$$

therefore,

$$\begin{aligned} & E\left[L(t+k) - L(t) | \Omega(t) \right] \\ \leq\ & \sum_{ijl} 2\lambda_{ijl} \left[k - S_{ijl}(t) \sum_{m=1}^{\eta k} E\left[\tau_{ijl}^{(m)}(t) \right] \right] W_{ijl}(t) + \\ & \sum_{ijl} \frac{\lambda_{ijl}}{C_l} \left[2C_l^2 k^2 + 2C_l^2 \eta k^2 \tau_{ijl}(t) \right] \\ \leq\ & \sum_{ijl} 2\lambda_{ijl} \left[k - S_{ijl}(t) \frac{\eta k}{\lambda_{ijl}} \right] W_{ijl}(t) + C \\ \leq\ & 2k \sum_{ijl} \left[\lambda_{ijl} W_{ijl}(t) - \eta W_{ijl}(t) S_{ijl}(t) \right] + C \end{aligned} \quad (19)$$

where $C = 2 \sum_{ijl} C_l \left[\lambda_{ijl} + \eta \right] k^2$. We further observe that for all $S_{ijl}(t) \neq 0$,

$$2S_{ijl}(t) = 2 > \sum_{il} \lambda_{ijl} + \sum_{jl} \lambda_{ijl} \quad (20)$$

Since FMWM removes two plains following each iteration, (20) holds for all iterations, and thus we conclude that $\sum_{ijl} \lambda_{ijl} W_{ijl}(t) < \sum_{ijl} 2 W_{ijl}(t) S_{ijl}(t)$, suggesting that for $\eta \geq 2$ there exists a value $\bar{\alpha} < 1$ for which $\sum_{ijl} \lambda_{ijl} W_{ijl}(t) < \bar{\alpha} \eta \sum_{ijl} W_{ijl}(t) S_{ijl}(t)$, Applying the latter to (19) yields

$$\begin{aligned} & E\left[L(t+k) - L(t) | \Omega(t) \right] \\ & \leq 2k(\bar{\alpha} - 1) W^{FMWM}(t) + C \end{aligned} \quad (21)$$

Thus, for all $W^{FMWM}(t) > \frac{C}{2k(1-\bar{\alpha})}$, we infer that $E\left[L(t+k) - L(t) | \Omega(t) \right] < 0$, which concludes the stability proof. ■

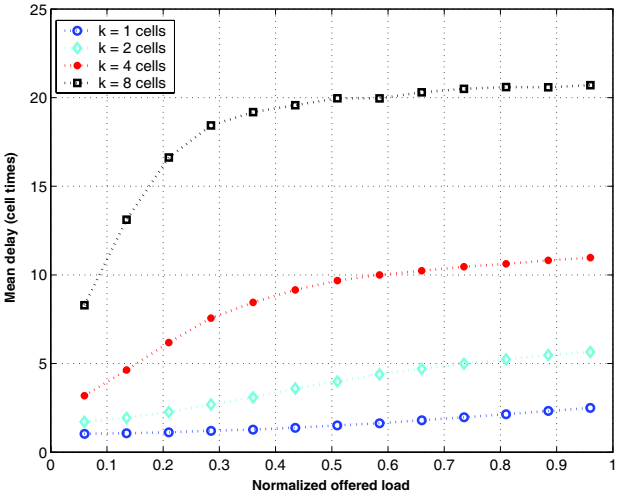

Fig. 3. MWMF/OCF mean queueing delay as a function of the frame size (k). The arrival process is Bernoulli i.i.d.

IV. Simulation Results

In order to evaluate the performance of the FMWM algorithm under different traffic conditions and interval durations, 4 simulation sets were conducted. In all simulations a 6-port switch was considered with a transfer speedup of 2. The x-axis denotes the normalized offered load, which is defined as the average number of cells arriving at each ingress port per time slot. In the first set of simulations, the arrival process was Bernoulli i.i.d. with uniformly distributed destination distribution. Fig. 3 shows the mean delay when employing FMWM/OCF with different switching frame sizes (k). As can be intuitively appreciated, the longer the frame the larger the mean delay, which stems from the fact that during many switching intervals less than k consecutive cells are being transmitted. Moreover, it is noted that larger frame sizes exhibit faster delay growth (steeper slope). This can be explained by the fact that once a matching matrix is generated, the unmatched VOQs will not transmit any cells during k time slots, yet they continue to buffer newly arriving cells (which contribute to the increase in the average queue waiting time).

Fig. 4 illustrates the mean delay under two-state Markov-modulated bursty traffic arrivals (i.e. ON/OFF process). The frame size here was 8 cells. It is interesting to note that the larger the mean burst size the lower the overall mean delay, which can be explained by the fact that bursts of packet arrivals will most likely be served during the same frame, which reduces the queueing latency.

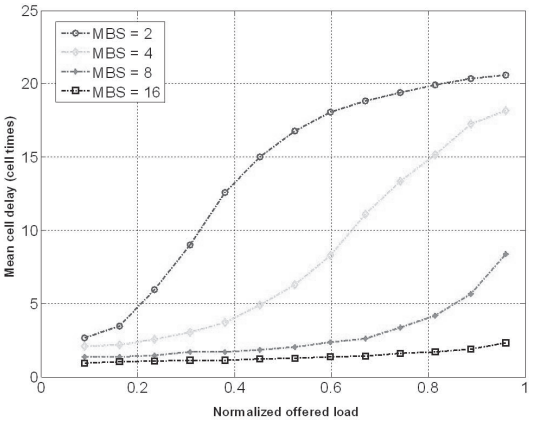

Fig. 4. MWMF/OCF mean delay for frame size of 8 cells and arrivals that are governed by an ON/OFF bursty process, with different mean burst sizes (MBS).

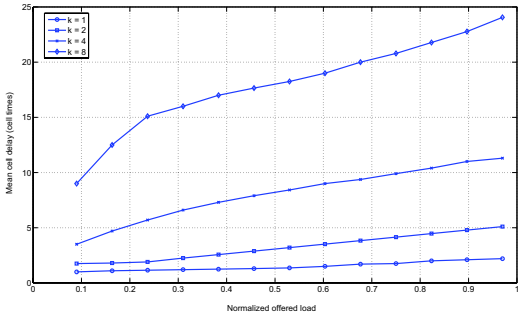

Fig. 5. Average cell delay for FMWM/OCF algorithm when arrivals are Bernoulli i.i.d. and distributed (non-uniformly) according to the Zipf law (with $r = 1$).

Real-life packet streams are distributed non-uniformly across the destinations, hence traffic tends to be focused on preferred, or popular, outputs. As means of evaluating the performance of the proposed architecture under non-uniform traffic conditions, we have selected the Zipf law as a destination distribution model [10], [12]. The Zipf law states that the frequency of occurrence of some events, as a function of the rank (m) which is determined by the above frequency of occurrence, is a power-law function, i.e. $P_k \approx 1/k^m$. It has been shown that many natural phenomena, such as Web access statistics, company sizes and biomolecular sequences, all obey the Zipf law with the order being close to 1 [12]. The probability that an arriving cell is heading to destination k was thus modeled by [4]

$$Zipf_m(k) = \lambda_m(k) = \frac{k^{-m}}{\sum_{j=0,1,..N} j^{-m}}. \qquad (22)$$

While $m = 0$ corresponds to uniform distribution, as m increases the distribution becomes more biased towards preferred destinations. Fig. 5 depicts the average delay for Zipf distributed packet streams with Bernoulli arrival characteristics, as a function of the offered traffic load.

The third set of simulations was targeted at examining the impact of multiple classes of service on the delay performance. Linear priority coefficients were used, such that $W_{ijl}(t) = l \times T_{ijl}(t)$. Once again, a 6 port switch with 3 classes of service was examined. The arrival processes was the same as before, while traffic was uniformly distributed across the classes of service. The frame size was set to 8 cells. As can be observed in Fig. 6, the higher the priority level the lower the mean delay experienced by arriving packets, a result of the fact that the scheduler tends to select the higher weights during each iteration. The disparity between the different classes of service can be controlled by tuning the priority coefficients. The theory presented in this paper clearly states that as long as the class coefficients are non-zero, the system is guaranteed to be stable.

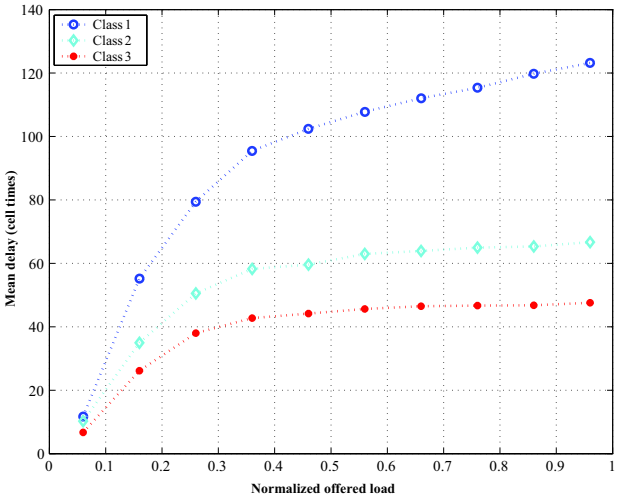

Fig. 6. FMWM/OCF mean delay for multiple classes of service. Classes of service are differentiated via linear priority coefficients.

It is interesting to note that although the average delay experienced by traffic associated with each class of service vary significantly, the average delay of all traffic (combining the three classes of service) remains constant. This can be intuitively appreciated due to the fact that the aggregation all the traffic streams is identical, regardless of the class of service distribution employed.

V. Conclusions

This paper studies the frame-based maximal weight matching algorithm, with OCF priorities, as a scalable scheduling scheme for large port-density input-queued switches. Through the use of Lyapunov functions, it has been shown that a transfer speedup of 2 is sufficient to guarantee stability for architectures hosting both single and multiple classes of service. The need for transfer speedup, as opposed to the scheduling speedup considered in the literature, renders the approach highly attractive from an implementation perspective. This is mainly due to the difficulty of completing the scheduling process within a single cell time, which is completely alleviated by the proposed framework. Moreover, the frame-based analysis presented here can be broadened to address other input-queued switching architectures and scheduling algorithms.

REFERENCES

[1] A. Bianco, M. Franceschinis, S. Ghisolfi, A. M. Hill, E. Leonardi, F. Neri, and R. Webb, "Frame-based matching algorithms for input-queued switches," in *Proc. IEEE High Performance Switching Routing Symp.*, May 2002, pp. 69–76.

[2] J. Dai and B. Prabhakar, "The throughput of data switches with and without speedup," *IEEE INFOCOM*, Mar. 2000, pp. 556–564.

[3] I. Elhanany and M. Hamdi, *High Performance Packet Switching Architectures*. London, UK: Springer-Verlag, 2006.

[4] I. Elhanany and D. Sadot, "DISA: A robust scheduling algorithm for scalable crosspoint-based switch fabrics," *IEEE J. Select. Areas Commun.*, vol. 21, no. 4, pp. 535–545, May 2003.

[5] Z. Guo and R. Rojas-Cessa, "Framed round-robin arbitration with explicit feedback control for combined input-crosspoint buffered packet switches," in *IEEE International Conf. Commun.*, June 2006, vol. 1, pp. 97–102.

[6] I. Keslassy, R. Zhang-shen, and N. McKeown, "Maximum size matching is unstable for any packet switch," *IEEE Commun. Lett.*, vol. 7, no. 10, pp. 496–498, Oct. 2003.

[7] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE Trans. Commun.*, vol. 47, no. 8, pp. 1260–1267, Aug. 1999.

[8] A. Mekkittikul and N. McKeown, "A starvation-free algorithm for achieving 100% throughput in an input-queued switch," in *Proc. ICCCN*, Oct. 1996, pp. 226–231.

[9] S. Mneimneh and K.-Y. Siu, "On achieving throughput in an input-queued switch," *IEEE/ACM Trans. Networking*, vol. 11, no. 5, pp. 858–867, 2003.

[10] A. L. Montgomery and C. Faloutsos, "Identifying web browsing trends and patterns," *Computer*, vol. 34, no. 7, pp. 94–95, 2001.

[11] Y. Tamir and H. C. Chi, "Symmetric crossbar arbiters for VLSI communication switches," *IEEE Trans. Parallel Distrib. Syst.*, vol. 4, no. 1, pp. 13–27, 1993.

[12] C. Williamson, "Internet traffic measurement," *IEEE Internet Comput.*, vol. 5, no. 6, Dec. 2001.