Maximum Size Matchings and Input Queued Switches[*]

Sundar Iyer, Nick McKeown
Computer Systems Laboratory, Stanford University,
Stanford, CA 94305-9030
{sundaes, nickm}@stanford.edu

Abstract — *Simulation results suggest that a maximum size matching (MSM) algorithm will lead to 100% throughput for uniform Bernoulli i.i.d traffic. Previous analysis on the throughput of MSM algorithms enforce deterministic constraints on the joint arrival traffic. In this paper we explore MSM algorithms under Bernoulli arrivals. We show that when the arrival traffic is admissible and cells are scheduled in batches, a sub-class of MSMs called the Critical Maximum Size Matching achieves 100% throughput under (uniform and non-uniform) Bernoulli i.i.d arrivals. Further, we show that — with batch scheduling — all MSMs achieve 100% throughput under Bernoulli i.i.d. uniform load.*

I. Introduction

A commonly used switching fabric in high speed packet switches (e.g. Internet routers, ATM switches) is a crossbar with input queues (IQ) to hold packets during times of congestion. A crossbar for an N port switch is constrained to schedule a matching i.e. it can send at most one packet from an input port and receive at most one packet at an output port in a single time slot.[1] It is known that such IQ crossbar switches suffer from throughput limitations called head-of-line (HoL) blocking which can limit the throughput of the switch to about 58% [1]. So it is common for IQ switches to maintain a separate queue for cells at each input destined to every output [2], commonly called VOQs, which eliminates HoL blocking. There exist a number of techniques which allow IQ switches with VOQs to obtain 100% throughput. It is desirable to achieve 100% throughput since it allows a network operator to efficiently utilize the expensive link bandwidth. For example, Chang et al. [3] and Altman et al. [4] showed (using the results of Birkhoff [5] and von Neumann [6]) that the crossbar can be scheduled by a fixed sequence of matchings (called frames) such that the IQ switch can achieve 100% throughput for any admissible arrival pattern.[2] These are similar to time division multiplexing (TDM) techniques in the switching literature [7]. However frame scheduling requires prior knowledge of the arrival traffic pattern and the switch needs to maintain a potentially long sequence of matchings.

[1] We will assume that time is slotted and at most one cell (of fixed size) can arrive at each of the N input ports of a switch in each time slot.

[2] Traffic is called admissible if no input or output is oversubscribed.

[*] This work was funded in part by the Stanford Networking Research Center.

Here we are interested in online algorithms which attempt to schedule traffic by computing a matching every time slot. In [8][9][10], it was shown that a maximum weight matching (MWM) algorithm, which has a time complexity of $O(N^3)$, (where the weights could be the queue lengths of the individual VOQs) can give 100% throughput for i.i.d. traffic.

In this paper we're interested in maximum size (or cardinality) matchings (MSM) which have time complexity $N^{2.5}$ [11]. We might expect MSM to achieve high throughput because it maximizes the size of the matching (and hence instantaneous throughput) in each time slot. However it is known that if ties are broken randomly, MSM does not achieve 100% throughput for all admissible Bernoulli traffic patterns [8]. Not all MSM algorithms suffer from loss of throughput. In [12] it is shown that the longest port first (LPF) algorithm (an MSM algorithm that uses weights to break ties) achieves 100% throughput for Bernoulli arrivals.

An open question about the MSM algorithm is: Can MSM give 100% throughput under *uniform* Bernoulli arrivals (i.e. when the destination of each arriving cell is picked uniformly and at random from among all outputs)? Simulation suggests that this is true, but to our knowledge no proof has been reported.

In [13], Weller and Hajek give a detailed analysis on the stability of online matching algorithms (including MSM) using frame scheduling. They use an (α, S) traffic model, where no more than $0 \le \alpha \le 1$ cells may arrive at any input or destined to any output in any S consecutive time slots. However the (α, S) traffic imposes deterministic constraints on the joint arrival sequences of packets arriving at all inputs destined to different outputs.

In this paper, we extend some results in [13] by considering stochastic (Bernoulli) arrivals. We show that using a slight modification of frame scheduling (called batch scheduling), a class of MSM algorithms (called Critical Maximum Size Matching, or CMSM), can achieve 100% throughput for (uniform or non-uniform) Bernoulli i.i.d. traffic, in addition to (α, S) arrival traffic. Second, we will show that if batch scheduling is used, any MSM algorithm can achieve 100% throughput under Bernoulli i.i.d. uniform load.

II. MSM With Batch Scheduling Gives 100% Throughput

A. Batch Scheduling

Previous work on frame scheduling assumes the frame size is fixed; we will analyze the switch under batch scheduling (proposed by Dolev and Kesselman [14]) in which the frame size varies with the arrival traffic and scheduling algorithm used.[3] In batch scheduling, all arriving cells are classified into one of two strict priority levels, p_1 and p_2. Each VOQ is replaced by two queues, one per priority level. An arriving cell is queued at priority p_2 if there

[3] The terms "batch scheduling" and "frame scheduling" have been used inter-changeably in the literature. In this paper "frame scheduling" means the frame size is fixed, whereas with "batch scheduling" it may vary.

are any cells (in any VOQ of the switch) at priority p_1. Since p_1 cells have strict priority over p_2 cells, at any given time t, only cells at p_1 will be considered for departure. If at some time all the cells at p_1 have been scheduled, and all VOQs at p_1 become empty, then all the cells at priority p_2 are immediately transferred to the p_1 queues. The set of transferred cells is called a *batch*. Newly arriving cells are stored in the now empty p_2 queues and will not be available to the scheduling algorithm until the current batch has departed. The batch number is referred to by its index k. We shall denote by B_k the time taken to schedule batch k. Each batch may contain multiple cells from a given input to an output. In what follows, we will represent a batch by a weighted bipartite graph G, with edge weights $w_k(i, j)$ representing the number of cells at input i destined to output j in batch k.

B. Preliminaries

Definition 1: *Degree* $d_{v,k}$*: The degree of an input (output) vertex* v*,* $d_{v,k}$*, is defined as the number of cells destined from input* v *(to output* v*) in batch* k*. i.e.* $d_{v,k} = \sum_{j=1}^{N} w_k(v, j)$*, if* v *is an input, and* $d_{v,k} = \sum_{i=1}^{N} w_k(i, v)$*, if* v *is an output.*

Definition 2: *Maximum Degree* D_k*: The maximum degree of batch* k*,* D_k *is the maximum degree amongst all input or output vertices in the weighted bipartite request graph* G *for batch* k*. i.e.* $D_k = \max_{\forall v \in G} d_{v,k}$.

It is a well known that any request graph with maximum degree M can be scheduled in M time slots [15]. This leads to the definition of the critical maximum size matching.

Definition 3: *Critical Maximum Size Matching (CMSM)**: A critical maximum size matching (CMSM) algorithm is a MSM that schedules batches of maximum degree* M *in* M *timeslots.*

The construction of a CMSM for any batch is shown in [13] (where it also noted that not all MSMs belong to the class of CMSMs).

Definition 4: *Weakly Stable:* *A system of queues* X_n *is said to be weakly stable if, for every* $\varepsilon > 0$*, there exists a* $D > 0$ *such that,* $\lim_{n \to \infty} P\{X_n > D\} < \varepsilon$*, (where* n *denotes time slots).*

Definition 5: *Strongly Stable:* *A system of queues* X_n *is said to be strongly stable, if it is weakly stable, and the limit,* $\lim_{n \to \infty} E(X_n)$ *is finite.*

Definition 6: ***100% Throughput: (Strong Sense)***[4]*: A service discipline D which services a system of queues X_n is said to give 100% throughput, if X_n is strongly stable.*

Theorem 1: *Consider a random variable whose evolution is described by a discrete time markov chain (DTMC) which is aperiodic and irreducible with state vector $Y_n \in \mathbf{N}^M$. Suppose that a lower bounded non-negative function $V(Y_n)$, called a Lyapunov function, V: $\mathbf{N}^M \to \mathbf{R}$ exists and suppose that A is a finite subset of $\mathbf{N}^M$, such that $V(Y_n) \le B' \Rightarrow Y_n \in A$, $B' \in \mathbf{R}^+$. Then if $E[V(Y_{n+1}) | Y_n] < \infty$, $\forall Y_n$ and there exist $\gamma \in \mathbf{R}^+$, $B \in \mathbf{R}^+$ such that*

$$E[V(Y_{n+1}) - V(Y_n) | Y_n] < -\gamma V(Y_n),\ \forall \|Y_n\| > B \tag{1}$$

then all states of the DTMC are positive recurrent and the process $V(Y_n)$ is strongly stable i.e. $\lim_{n \to \infty} E[V(Y_n)]$ is finite.

Proof: This is a straightforward extension of Foster's criteria and follows from [16][17][18]. ❒

C. Stability of CMSM under Bernoulli Traffic

In this section we assume that arrivals to input $i \in \{1, 2, \ldots, N\}$ are Bernoulli i.i.d. with rate α_i, and are destined to each output $j \in \{1, 2, \ldots, N\}$ with probability $\frac{\alpha_{i,j}}{\alpha_i}$. We will denote the arrival matrix as,[5]

$$A \equiv [\alpha_{i,j}],\quad \text{where: } \alpha_j = \sum_{i=1}^{N} \alpha_{i,j} < 1,\ \alpha_i = \sum_{j=1}^{N} \alpha_{i,j} < 1,\ \alpha_{i,j} \ge 0$$

Theorem 2: *The process B_k^2 is strongly stable under batch scheduling with the CMSM, if input traffic is admissible and Bernoulli i.i.d. uniform.*

Proof: Consider the evolution of the process B_k (the time taken to schedule batch k). B_k is an (aperiodic and irreducible) discrete time[6] markov chain, since

$$\begin{aligned} &P\{B_{k+1} = j | B_0 = i_0, B_1 = i_1, \ldots, B_k = i_k\} \\ &= P\{B_{k+1} = j | B_k = i_k\} \end{aligned}$$

We define a quadratic Lyapunov function, $V(B_k) = B_k^2$. Because input traffic is uniform and admissible, $\alpha_{i,j} = \frac{\alpha}{N}, \forall (i,j) \in G$, with $\alpha < 1$. Hence, for all input ports $v \in G$, $\boldsymbol{E}[d_{v,k+1} | B_k] = \alpha B_k$, and for all output ports $v \in G$, $\boldsymbol{E}[d_{v,k+1} | B_k] = \sum_{i \in \{1, \ldots, N\}} \frac{\alpha}{N} B_k = \alpha B_k$. Hence,

[4] Other definitions of 100% throughput exist in the literature.

[5] In what follows we shall assume that the arrival rate of cells at a particular input or output, $\alpha_i, \alpha_j < 1$ for the traffic to be admissible. This is contrast to the traffic model used in [13] where $\alpha \le 1$.

[6] Strictly speaking 'time' here does not refer to a process with constant increments as in the usual sense.

$$\boldsymbol{E}[d_{v,\,k+1}|B_k] = \alpha B_k,\ \forall v \in G.$$

From the Chernoff bound, and for any $\delta > 0$ we get $\forall v \in G$

$$P\{d_{v,\,k+1} > (1+\delta)\alpha B_k | B_k\} < \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\alpha B_k} \tag{2}$$

We would like to bound the time taken to schedule batch $k+1$ i.e. B_{k+1}, given knowledge of B_k. By definition, a CMSM schedules batch k in D_k slots, and so we can bound B_{k+1} by bounding D_{k+1}. Since the distribution of D_{k+1} is bounded by the maximum degree amongst each of the N inputs and N outputs, we can use Equation (2) and the union bound to write,

$$P\{D_{k+1} \le (1+\delta)\alpha B_k | B_k\} \tag{3}$$

$$= P\left\{\bigcap_{v \in G} \{d_{v,\,k+1} \le (1+\delta)\alpha B_k | B_k\}\right\} = 1 - P\left\{\bigcup_{v \in G} \{d_{v,\,k+1} > (1+\delta)\alpha B_k | B_k\}\right\}$$

$$\ge 1 - \sum_{v \in G} P\{d_{v,\,k+1} > (1+\delta)\alpha B_k | B_k\} > 1 - 2N\left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\alpha B_k} \equiv Q.$$

Choose an $\varepsilon > 0$ such that $\varepsilon < 1 - \alpha$; in particular let $\varepsilon = (1-\alpha)/2$. We are interested in making $(1+\delta)\alpha$ less than $(1-\varepsilon) = (1+\alpha)/2$. So we choose $\delta > 0$ such that, $\delta < [(1-\varepsilon)/\alpha] - 1$, i.e. $\delta < (1-\alpha)/2\alpha$. Choosing $\delta = (1-\alpha)/4\alpha$ we can write Equation (3) as,

$$P\{D_{k+1} < (1+\alpha)B_k/2 | B_k\} > Q.$$

Since $B_{k+1} = D_{k+1}$, we have $P\{B_{k+1} < (1+\alpha)B_k/2 | B_k\} > Q$. Now, let event $A \equiv \{B_{k+1} < (1+\alpha)B_k/2 | B_k\}$ and let $A^c \equiv \{B_{k+1} \ge (1+\alpha)B_k/2 | B_k\}$. Then $P\{A\} > Q$ and $P\{A^c\} \le 1 - Q$, and so

$$E[V(B_{k+1})|B_k] = P\{A\}E[V(B_{k+1})|(A, B_k)] + P\{A^c\}E[V(B_{k+1})|(A^c, B_k)].$$

Since N independent Bernoulli processes can generate at most N cells in a time slot we have $B_{k+1}|B_k \le NB_k$, which leads to $E[V(B_{k+1})|(A^c, B_k)] \le N^2 B_k^2 \le N^2 V(B_k)$, and hence

$$E[V(B_{k+1})|B_k] < Q\left(\frac{1+\alpha}{2}\right)^2 V(B_k) + (1-Q)N^2 V(B_k)\ \ .$$

Thus $E[V(B_{k+1})|B_k]$ is bounded by a convex combination of a number which is less than $V(B_k)$ i.e. $(1+\alpha)^2 V(B_k)/4$ and a number which is greater than $V(B_k)$ i.e. $N^2 V(B_k)$. Choosing large enough Q can make the combination strictly less than $(1-\gamma)V(B_k)$, where $0 < \gamma < 1$. Hence,

$$Q\left(\frac{1+\alpha}{2}\right)^2 V(B_k) + (1-Q)N^2 V(B_k) < (1-\gamma)V(B_k)\,, \tag{4}$$

which can be re-written as $Q > \frac{N^2 - 1 + \gamma}{N^2 - a^2}$, where $a = \left(\frac{1+\alpha}{2}\right)^2$.

Observe from Equation (3) that Q is always less than one. We can choose γ above so that Q is less than 1. Hence, $0 < \gamma < 1 - a^2$, which implies that $0 < \gamma < (3 - 2\alpha - \alpha^2)/4$. We will choose $\gamma = (3 - 2\alpha - \alpha^2)/8$, so that the convex combination is less than $(1-\gamma)V(B_k) = (5 + 2\alpha + \alpha^2)V(B_k)/8$.

Note that since we have fixed δ; and α, N are constants, Q is solely a function of B_k. Also note as B_k increases, $Q \to 1$ and Q is a strictly increasing function of B_k. Hence the inequality in Equation (3) is satisfied for some large enough value of $B_k > C_1$, where C_1 is a constant. Then the quadratic Lyapunov function $V(.)$ satisfies Equation (1) and we can write,

$$E[V(B_{k+1}) - V(B_k)|B_k] < -\left(\frac{3 - 2\alpha - \alpha^2}{8}\right)V(B_k)\,;\forall B_k, (B_k > C_1)\,.$$

Also since $E[V(B_{k+1})|B_k] \le N^2 B_k < \infty$ all the conditions in Theorem 1 are all satisfied. Hence the process $V(B_k)$ is strongly stable (over batch index k). From Definition 5, $\lim_{k \to \infty}\{E[B_k^2]\} < \infty$. ❐

Theorem 3: *CMSM gives 100% throughput under batch scheduling, if the input traffic is admissible and Bernoulli i.i.d. uniform.*

Proof: Let many-to-one function $k(t)$ denote the index of the batch that is being scheduled at time t. We'll define $R(k(t))$ to be the proportion of time that the batch with index $k(t)$ is seen in a period of time T. Specifically we define $R(k(t)) = B_{k(t)}/T$.[7]

First, we'll show that the process $B_{k(t)}$ is strongly stable over time. Since the DTMC is ergodic, we can write,

$$\lim_{T \to \infty}\{E[B_{k(T)}]\} = \lim_{T \to \infty}\left\{\sum_{t=0}^{T} \frac{B_{k(t)}}{T}\right\} \tag{5}$$

Also,

[7] Note that this definition overestimates the proportion of time that the last batch $k(T)$ is seen in a period of time T, since it is possible that the last batch has not been completely scheduled at the end of time T. We take care of this boundary condition later.

$$\sum_{k(t)=0}^{k(T)} R(k(t))B_{k(t)} - \left(\frac{B_{k(T)}}{T}\right)B_{k(T)} < \sum_{t=0}^{T} \frac{B_{k(t)}}{T} < \sum_{k(t)=0}^{k(T)} R(k(t))B_{k(t)} + \left(\frac{B_{k(T)}}{T}\right)B_{k(T)} \tag{6}$$

$$\Rightarrow \lim_{T\to\infty} \sum_{k(t)=0}^{k(T)} R(k(t))B_{k(t)} - \frac{{B^2}_{k(T)}}{T} \le \lim_{T\to\infty} \sum_{t=0}^{T} \frac{B_{k(t)}}{T} \le \lim_{T\to\infty} \sum_{k(t)=0}^{k(T)} R(k(t))B_{k(t)} + \frac{{B^2}_{k(T)}}{T}.$$

Also Theorem 2 implies that $\lim\limits_{T\to\infty} \frac{{B^2}_{k(T)}}{T} \to 0$, and so we get from Equation (6),

$$\lim_{T\to\infty}\left\{\sum_{t=0}^{T} \frac{B_{k(t)}}{T}\right\} = \lim_{T\to\infty}\left\{\sum_{k(t)=0}^{k(T)} R(k(t))B_{k(t)}\right\} =$$

$$\lim_{T\to\infty}\left\{\sum_{k(t)=0}^{k(T)} \frac{B_{k(t)}B_{k(t)}}{T}\right\} = \lim_{T\to\infty}\left\{\frac{k(T)}{T}\left(\sum_{k(t)=0}^{k(T)} \frac{B_{k(t)}B_{k(t)}}{k(T)}\right)\right\}.$$

But since $\forall T,\ k(T)/T < 1$ and using Equation (5) we get,

$$\lim_{T\to\infty}\{E[B_{k(T)}]\} = \lim_{T\to\infty}\left\{\sum_{t=0}^{T} \frac{B_{k(t)}}{T}\right\} = \tag{7}$$

$$\lim_{T\to\infty}\left\{\frac{k(T)}{T}\left(\sum_{k(t)=0}^{k(T)} \frac{B_{k(t)}B_{k(t)}}{k(T)}\right)\right\} \le \lim_{T\to\infty}\left\{\sum_{k(t)=0}^{k(T)} \frac{B_{k(t)}B_{k(t)}}{k(T)}\right\}.$$

Also, since $\lim\limits_{k\to\infty}\{E[B_k^2]\} < \infty$ then $\lim\limits_{k\to\infty}\{E[B_k]\} < \infty$. This means that $T \to \infty \Rightarrow k(T) \to \infty$. So we can write,

$$\lim_{T\to\infty}\left\{\sum_{k(t)=0}^{k(T)} \frac{B_{k(t)}B_{k(t)}}{k(T)}\right\} = \lim_{k(T)\to\infty}\left\{\sum_{k(t)=0}^{k(T)} \frac{B_{k(t)}B_{k(t)}}{k(T)}\right\} \quad .$$

Changing the index $k(T)$ back to T we can write,

$$\lim_{k(T)\to\infty}\left\{\sum_{k(t)=0}^{k(T)} \frac{B_{k(t)}B_{k(t)}}{k(T)}\right\} = \lim_{T\to\infty}\left\{\sum_{t=0}^{T} \frac{B_t B_t}{T}\right\} = \lim_{T\to\infty}\left\{\sum_{k=0}^{T} \frac{B_k B_k}{T}\right\} = \lim_{T\to\infty}\left\{\sum_{k=0}^{T} \frac{{B^2}_k}{T}\right\} \quad .$$

But we can write the Cesaro average as, $\lim\limits_{T\to\infty}\left\{\sum\limits_{k=0}^{T} \frac{{B^2}_k}{T}\right\} = \lim\limits_{T\to\infty}\{E({B^2}_T)\}$, and since from Theorem 2, $\lim\limits_{T\to\infty}\{E({B^2}_T)\} < \infty$, we can substitute in Equation (7) to get

$$\lim_{T\to\infty}\{E[B_{k(T)}]\} = \lim_{T\to\infty}\left\{\sum_{t=0}^{T} \frac{B_{k(t)}}{T}\right\} < \infty . \tag{8}$$

We are now ready to show that the switch gives 100% throughput. Let Q_t denote the size of all the input queues at time t. We will show that Q_t is strongly stable over time. We know that, $Q_t < N\{B_{k(t)} + B_{k(t)+1}\}$. Also since $B_{k(t)+1} < NB_{k(t)}$, we have, $Q_t < (N^2 + N)\{B_{k(t)}\}$.

Now let us consider the Cesaro average of the process Q_t,

$$\lim_{T \to \infty}\left\{\sum_{t=0}^{T} \frac{Q_t}{T}\right\} < \lim_{T \to \infty}\left\{\sum_{t=0}^{T} \frac{(N^2+N)B_{k(t)}}{T}\right\} = (N^2+N)\lim_{T \to \infty}\left\{\sum_{t=0}^{T} \frac{B_{k(t)}}{T}\right\}.$$

From Equation (8) it follows that the Cesaro mean is finite i.e. $\lim_{T \to \infty}\left\{\sum_{t=0}^{T} \frac{Q_t}{T}\right\} < \infty$. Hence the Markov chain defining the process Q_t is ergodic and we can write,

$$\lim_{T \to \infty}\{E[Q_T]\} = \lim_{T \to \infty}\left\{\sum_{t=0}^{T} \frac{Q_t}{T}\right\} < \infty .$$

Since $\lim_{T \to \infty}\{E[Q_T]\} < \infty$ the system of queues $Q(t)$ is strongly stable, and the switch has 100% throughput. ❐

Theorem 4: *CMSM gives 100% throughput under batch scheduling, for any Bernoulli i.i.d admissible arrival process.*

Proof: The proof is similar to Theorem 2. Instead of using the uniform arrival rate α for all inputs and outputs as in Equation (2) and Equation (3) in Theorem 2, we replace these by the individual Bernoulli arrival rates α_i, at each input i or the rates α_j to each output j for the $2N$ vertices. ❐

D. Stability of MSM under uniform load using batch scheduling

Consider the distribution of the VOQ with minimum length in batch $k+1$, i.e. we define

$$w^*_{k+1} = \min_{\forall (i,j) \in \{1, 2, \ldots, N\}} w_{k+1}(i,j) .$$

Lemma 1: *If the minimum size of the VOQ in batch $k+1$ is w^*_{k+1} and the maximum degree of the request graph G for batch $k+1$ is D_{k+1}, then any MSM will serve batch $k+1$ within $2D_{k+1} - w^*_{k+1}N$ time slots.*

Proof: Consider the request graph G for batch $k+1$. Split G into two request graphs G_1 and G_2. Let G_1 be a perfect graph i.e. a graph such that all VOQs have length w^*_{k+1}. Let G_2 be a request graph corresponding to all the remaining cells in batch $k+1$ not contained in G_1. The maximum degree of G_2 is bounded by $D_{k+1} - w^*_{k+1}N$. As shown in [14], any MSM can schedule G_1 using $w^*_{k+1}N$ matchings of size N. Also, since any MSM is also a maximal matching, any MSM can schedule G_2 with $2(D_{k+1} - w*_{k+1}N)$ matchings. Thus the total time taken to service batch $k+1$ is $2(D_{k+1} - w*_{k+1}N) + w*_{k+1}N = 2D_{k+1} - w*_{k+1}N$. ❐

Theorem 5: *MSM gives 100% throughput under batch scheduling, if the input traffic is admissible and Bernoulli i.i.d. uniform.*

Proof: Similar to Theorem 2, we consider the evolution of the DTMC process B_k, and the same quadratic Lyapunov function, $V(B_k) = B_k^2$. Recall that $w_k(i,j)$ represents the number of cells from input i destined to output j in batch k. Hence,

$$\boldsymbol{E}[w_{k+1}(i,j) | B_k] = \frac{\alpha}{N} B_k,\ \forall i,j \in \{1, 2, \ldots, N\}.$$

From the Chernoff bound, and for any $\delta_2 > 0$ we get $\forall i, j \in \{1, 2, \ldots, N\}$

$$P\left\{ w_{k+1}(i,j) < (1-\delta_2)\frac{\alpha}{N} B_k \middle| B_k \right\} < e^{-\frac{\alpha}{N} B_k \frac{\delta_2^{\ 2}}{2}}. \qquad (9)$$

Now we use Equation (9) to get,

$$P\left\{ w*_{k+1} \geq (1-\delta_2)\frac{\alpha}{N} B_k \middle| B_k \right\}$$

$$= P\left\{ \bigcap_{\forall(i,j)} \left\{ w_{k+1}(i,j) \geq (1-\delta_2)\frac{\alpha}{N} B_k \middle| B_k \right\} \right\} = 1 - P\left\{ \bigcup_{\forall(i,j)} \left\{ w_{k+1}(i,j) < (1-\delta_2)\frac{\alpha}{N} B_k \middle| B_k \right\} \right\}$$

$$\geq 1 - \sum_{\forall(i,j)} P\left\{ w_{k+1}(i,j) < (1-\delta_2)\frac{\alpha}{N} B_k \middle| B_k \right\} > 1 - N^2 \left(e^{-\frac{\alpha}{N} B_k \frac{\delta_2^{\ 2}}{2}} \right) \equiv Q_1.$$

(10)

Also from Equation (3) replacing the symbol δ by δ_1 we get for any $\delta_1 > 0$,

$$P\{D_{k+1} \leq (1+\delta_1)\alpha B_k | B_k\} > 1 - 2N \left(\frac{e^{\delta_1}}{(1+\delta_1)^{(1+\delta_1)}} \right)^{\alpha B_k} \equiv Q_2. \qquad (11)$$

Define the two events $E_1 \equiv \left\{ w*_{k+1} \geq (1-\delta_2)\frac{\alpha}{N} B_k \middle| B_k \right\} \equiv \{ w*_{k+1}N \geq (1-\delta_2)\alpha B_k | B_k \}$ and, $E_2 \equiv \{D_{k+1} \leq (1+\delta_1)\alpha B_k | B_k\}$. Note that by definition $P\{E_1\} > Q_1$ and $P\{E_2\} > Q_2$. Let $E \equiv B_{k+1} \leq (1 + 2\delta_1 + \delta_2)\alpha B_k | B_k$. From Lemma 1 we have $E_1 \cap E_2 \Rightarrow E$. Hence, we can write the following weak inequality,

$$P\{E\} \geq P\{E_1 \cap E_2\} = 1 - P\{E_1^c \cup E_2^c\}$$
$$\geq 1 - P\{E_1^c\} - P\{E_2^c\} > Q_1 + Q_2 - 1.$$

Hence, $P\{B_{k+1} \leq (1 + 2\delta_1 + \delta_2)\alpha B_k | B_k\} > Q_1 + Q_2 - 1$. We shall choose $\delta_1 > 0$ and $\delta_2 > 0$ such that $(1 + 2\delta_1 + \delta_2)\alpha < 1 - \varepsilon$, where $0 < \varepsilon < 1 - \alpha$. In particular choose $\varepsilon = (1 - \alpha)/2$. If we set $\delta_1 = \delta_2 = \delta^*$, then $\delta^* < (1 - \alpha)/6\alpha$. To satisfy this inequality choose $\delta_1 = \delta_2 = \delta^* = (1 - \alpha)/12\alpha$. This leads to

$$P\{B_{k+1} < (1 + \alpha)B_k/2 | B_k\} > Q_1 + Q_2 - 1 \equiv Q'. \tag{12}$$

Let event $A \equiv B_{k+1} < (1 + \alpha)B_k/2 | B_k$ and let $A^c \equiv B_{k+1} \geq (1 + \alpha)B_k/2 | B_k$. And so

$$E[V(B_{k+1}) | B_k] = P\{A\}E[V(B_{k+1}) | (A, B_k)] + P\{A^c\}E[V(B_{k+1}) | (A^c, B_k)]. \tag{13}$$

We know that $V(B_{k+1}) | B_k \leq NB_k$ which, with Equation (12) and Equation (13) leads to,

$$E[V(B_{k+1}) | B_k] < Q'\left(\frac{1 + \alpha}{2}\right)^2 V(B_k) + (1 - Q')N^2 V(B_k). \tag{14}$$

Similar to the reasoning in Theorem 2, $E[V(B_{k+1}) | B_k]$ is a convex combination of $(1 + \alpha)^2 V(B_k)/4$ and $N^2 V(B_k)$. Again we will choose, $\gamma = (3 - 2\alpha - \alpha^2)/8$ such that the convex combination is less than $(1 - \gamma)V(B_k) = (5 + 2\alpha + \alpha^2)V(B_k)/8$. This is true if we choose Q' so as to satisfy the inequality $\frac{N^2 - 1 + \gamma}{N^2 - a^2} < Q' < 1$, where $a = \left(\frac{1 + \alpha}{2}\right)^2$.

As before, note that as B_k increases, $Q' \rightarrow 1$, and since Q' is a strictly increasing function of B_k, the inequality in Equation (3) would be true for some $B_k > C_2$, where C_2 is a constant. Then the quadratic Lyapunov function $V(.)$ would satisfy both conditions in Theorem 1. Similar to the reasoning in Theorem 2 (and Theorem 3), MSM gives 100% throughput under batch scheduling. ❐

III. CONCLUSIONS

Our work was motivated by simulation results that suggest that MSM (with ties broken randomly) will give 100% throughput for uniform Bernoulli i.i.d. arrivals. Previous work on the stability of MSM imposed deterministic constraints on the arrivals. We extended these results to show that the class of CMSM algorithms achieves 100% throughput under (uniform and non-uniform) Bernoulli i.i.d. arrivals, and — when using batch scheduling — the general class of MSM algorithms achieve 100% throughput, if the traffic is Bernoulli i.i.d. uniform. Our results don’t quite meet our original goal of proving this result for continuous scheduling i.e. without grouping arrivals into batches. This remains an open question.

IV. Acknowledgements

We would like to thank Balaji Prabhakar and Devavrat Shah for some useful discussions. We would also like to thank Michael Neely and Bruce Hajek for pointing out errors in an earlier version.

V. References

[1] M. Karol, M. Hluchyj, S. Morgan, “Input versus Output Queueing on a Space-Division Packet Switch”, *IEEE Trans. on Communications*, vol. COM-35, no. 12, December 1987, pp. 1347-1356.

[2] Y. Tamir and H. C. Chi, "Symmetric crossbar arbiters for VLSI communication switches", *IEEE Transactions on Parallel and Distributed Systems*, vol. 4, No. 1, pp. 13-27, Jan. 1993.

[3] C.-S. Chang, W.-J. Chen, and H.-Y. Huang, “On service guarantees for input buffered crossbar switches: A capacity decomposition approach by Birkhoff and von Neumann,” *In IEEE Infocom*, Tel Aviv, Israel, 2000.

[4] E. Altman, Z. Liu, R. Righter, “Scheduling of an input-queued switch to achieve maximal throughput”, Probability in the Engineering and Informational Sciences, pp. 327-334, vol. 14, 2000.

[5] G. Birkhoff, "Tres observaciones sobre el algebra lineal," Univ. Nac. Tucum an Rev. Ser. A, vol. 5, pp. 147--151, 1946.

[6] J. von Neumann, "A certain zero-sum two-person game equivalent to the optimal assignment problem, " Contributions to the Theory of Games, Vol. 2, pp. 5-12, Princeton University Press, Princeton, New Jersey, 1953.

[7] Inukai, T. (1979), “An efficient SS/TDMA time slot assignment algorithm,” *IEEE Transactions on Communications*, COM-27:1449-1455.

[8] N. McKeown, V. Anantharam, J. Walrand, “Achieving 100% Throughput in an input-queued switch,” *Proceedings of IEEE Infocom ‘96*, vol. 1, pp. 296-302, March 1996.

[9] L. Tassiulas, “Scheduling and performance limits of networks with constantly changing topology,” IEEE Trans. Inform. Theory, vol. 43, pp. 1067-1073, May 1997.

[10] J. Dai and B. Prabhakar, "The throughput of data switches with and without speedup," *in Proceedings of IEEE INFOCOM '00*, Tel Aviv, Israel, March 2000, pp. 556 -- 564.

[11] J. E. Hopcroft, R. M. Karp, “An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs”, SIAM Journal on Computing, 2(4):225-231, December 1973.

[12] Adisak Mekkittikul, and Nick McKeown, “A Practical Scheduling Algorithm to Achieve 100% Throughput in Input-Queued Switches." IEEE Infocom 98, Vol 2, pp. 792-799, April 1998, San Francisco.

[13] Timothy Weller, Bruce Hajek, “Scheduling nonuniform traffic in a packet-switching system with small propagation delay,” *IEEE/ACM Transactions on Networking* 5(6): 813-823, 1997.

[14] Shlomi Dolev and Alexander Kesselman, “Bounded latency scheduling scheme for ATM cells", Computer Networks, vol. 32, no. 3, pp.325-331, 2000.

[15] M. Hall, Jr., Combinatorial Theory, Waltham, MA, Blaisdell, 1969.

[16] L. Tassiulas and A. Ephremides, "Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks," IEEE Trans. on Tutomatic Control, 37, December 1992, pp. 1936-1949.

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

[18] G. Fayolle, “On random walks arising in queuing systems: ergodicity and transience via quadratic forms as lyapunov functions - Part I”, *Queueing Systems*, vol. 5, pp. 167-184, 1989.

[19] R. Motwani and P. Raghavan, "Randomized Algorithms", Published by Cambridge University Press, Cambridge UK and New York, 1995.