Analysis of Scheduling Schemes in IQ switches under uniform traffic

Varun Malhotra, Wooyul Lee

June 6, 2003

Contents

Abstract

In the design of switches, good scheduling algorithms play a very pivotal role in deciding the throughput of the switch and giving some delay guarantees for the packets. Although various algorithms which achieve 100% throughput have been proposed and analyzed, most of these scheduling algorithms are variants of Maximum-Weight Matching (MWM) algorithms in bipartite graphs. Different MWM algorithms use different criteria for assigning the weights to the edges using some parameters dependent on the queue lengths of the Virtual Output Queues (VOQs). Similar class of algorithms, Maximum Size Matching Algorithms, have also been studied. It has been shown that Maximum-Size matching (MSM) scheduling algorithm doesn't give 100% throughput under non-uniform traffic pattern [1].

There have been results which have used some specific MSM algorithms for scheduling traffic and which have been proved stable, i.e., they give 100% throughput. But the question of how it performs under uniform traffic remains open. Although simulations suggest the stability of MSM algorithms under uniform traffic but there have been no analytical results proving the same. We analysed some of the scheduling algorithms including MSM, in input queued (IQ) switches with VOQs under uniform traffic. We give a counter-example showing that MSM is unstable for adversarial, uniform, admissible traffic if the MSM schedules are chosen adversarially. We prove the stability of 2x2 switch under bernoulli, i.i.d. traffic. Furthermore, we show that stability of Wait-Until-Full algorithm under Bernoulli i.i.d. uniform traffic.

1 Introduction

We consider input queued switches with VOQs and study the throughput of these switches under uniform *admissible* traffic when MSM scheduling algorithms are used. We will introduce the *notations* used through out this text before proceeding further. We will assume that the time is *slotted*.

Let Q_{ij} denote the VOQ for output j at input i. We will use $Q_{ij}(n)$ to denote the backlog in the VOQ Q_{ij} at the end of time slot n. If $\lambda_{ij}(n)$ denote the number of arrivals in VOQ Q_{ij} at the beginning of time slot n and $S_{ij}(n)$ be the number of packets that are served in the same time slot from this VOQ ($0 \leq \lambda_{ij},\ S_{ij} \leq 1,\ \forall\ i,j$), then, the dynamics of this VOQ can be modeled as follows:

$$Q_{ij}(n) = [Q_{ij}(n-1) - S_{ij}(n)]^{+} + \lambda_{ij}(n)$$

For *admissible* traffic, the following inequalities hold true ($\forall i, j$):

$$\begin{array}{rcl} \sum_i \lambda_{ij}(n) & < & 1 \\ \sum_j \lambda_{ij}(n) & < & 1 \end{array}$$

Our goal is to analyze the behavior (throughput) of an input queued switch with VOQs under *uniform*, *admissible* traffic.

2 Prior work

Weller and Hajek [3] introduced a new model of packet arrival traffic and analyzed the throughput of simple packet switching systems under this model. This model constrained the number of packets to any input or for any output port over time periods of specified length.

Iyer and McKeown [2] extended their arguments for stochastic (*Bernoulli*) arrivals and showed the stability of a class of MSM algorithms (called *critical MSM*) under batch scheduling. Recently, Keslassy, Zhang and McKeown [4] showed MSM is unstable for any switch packet switch by proving the instability of `2x2` switch. We use similar arguments in our analysis. Wischik et al. [6] propose a model for analyzing scheduling algorithms based on MWM matching. Although we haven't been able to make direct correlation of their technique in our context, but the paper puts forth elegant mathematical machinery for analyzing packet switches.

3 Analysis Strategy

There are two sources of randomness in the problem of MSM scheduling under Bernoulli uniform traffic. One is the stochastic nature of arrivals and another is the random choice of MSM schedule if there exist more than one at any time instant. To get some insights into the problem, let us start off with a very simplified model wherein the *arrivals* are deterministic. We construct a traffic pattern wherein the *adversary* is free to choose the MSM schedules under admissible and deterministic arrival patterns and show that the switch is not *stable*. We state an example for a simple `3x3` switch (for clarity) and will discuss how to extend it for `NxN` switch.

3.1 Counter Example

We represent the arrival pattern in a time slot by a matrix M, where M_{ij} entry is blank if there is no arrival in Q_{ij} while it can contains a 1[1] if a packet arrived in the VOQ corresponding to that entry in that time slot. Shown below are the *snapshots* of arrival patterns for 4 consecutive time intervals.

$$\left[\begin{array}{c|c|c} 1 & & \\ - & - & - \\ (1) & & \\ - & - & - \\ & & (1) \end{array}\right] \Rightarrow \left[\begin{array}{c|c|c} & & \\ - & - & - \\ & (1) & \\ - & - & - \\ (1) & & \end{array}\right] \Rightarrow \left[\begin{array}{c|c|c} & (1) & \\ - & - & - \\ & & (1) \\ - & - & - \\ & & \end{array}\right] \Rightarrow \left[\begin{array}{c|c|c} & & (1) \\ - & - & - \\ & & \\ - & - & - \\ & (1) & \end{array}\right]$$

Thus, we note that each VOQ has exactly one arrival in these 4 time slots, which gives $\lambda_{ij} = \frac{1}{4}$, $\forall i, j \in \{1, 2, 3\}$. Moreover, the traffic is uniform and *admissible*. In this example, if the same pattern repeats and the *adversary* chooses the MSM at points where a choice has to be made, then, VOQ Q_{11} never gets served and it will grow unbounded.

This counter example can be easily extended to an `NxN` switch as follows. We will simply state the strategy for constructing adversarial traffic patterns in this scenario without going into formal proofs. For any VOQ Q_{ij}, let us define the set of all the VOQs Q_{ik} where $k \neq j$, as the **input conflicting set**(ICS) of Q_{ij}. Similarly, set of all VOQs Q_{kj} where $k \neq i$, as the **output conflicting set**(OCS) of Q_{ij}. Now, consider a sequence of consecutive $2(N-1)$ time slots. If the adversary wants to make VOQ Q_{11} overflow, construct an arrival traffic pattern that contains one and only one arrival in any of the VOQs in the ICS or OCS in these $2(N-1)$ slots while adhering to the constraints of *uniformity*. We can fill the slots for rest of the VOQs appropriately so that that no two arrivals conflict with each other in the same time slot. Say a packet arrives in Q_{11} in the first time slot[2]. Since, the adversary has the choice of choosing MSMs, she can always choose to serve a packet in one of the VOQs in the ICS or OCS of Q_{11} over the packets in Q_{11} thus causing it to grow unbounded if the same trend follows over an extended period of time. Thus, with $\lambda_{ij} = \frac{1}{2(N-1)}$ such an arrival pattern is uniform and admissible, however, the switch is unstable under adversarial choice of MSM schedules.

It is interesting to note that if we restrict the *adversary* and make a deterministic decision (like choosing higher numbered inputs/outputs over lower numbered inputs and outputs) while choosing between a set of MSMs where the deterministic strategy of choosing MSMs is not fair for all VOQs[3]. Hereagain, it is easy to construct a counter-example

[1] (1) tells which packets were served

[2] We have a conflict only in the first time slot

[3] In the counter-example, VOQ_{11} has lower priority over other VOQs

and show that the switch is not stable if we use some deterministic strategy of choosing MSMs which can lead to starvation of one of VOQs. However, an interesting question is to investigate the behavior when the MSMs are chosen randomly or in some deterministic way (similar to *iSlip*) which does not starve any specific VOQs.

4 Stability of 2x2 switch

The problem of analysing the switch using MSM scheduling algorithm becomes harder when the arrival process is a stochastic process. We analyzed the 2x2 switch and found it to be stable under *uniform*, Bernoulli *admissible* traffic. Although, we came up with arguments proving the stability of 2x2 switch, we used a lemma in [4] (pointed to us by one of our collegues) in the final proof. As far as we know the stability of 2x2 switch has not been reported by [4] and we extend their arguments by a corollary.

Lemma *Under Bernoulli admissible traffic,* a 2x2 *switch can have at most one unstable queue.*
Proof For the sake of brevity, we are not replicating the arguments here. For details, refer to [4].

Corollary *Under uniform Bernoulli admissible traffic,* a 2x2 *switch is stable.*
Proof Given that $\lambda_{ij} = \lambda, \forall i, j \in \{1, 2\}$ and $\lambda < \frac{1}{2}$ for admissible arrival traffic.

Let us assume the contrary, i.e. one of the VOQs (say Q_{11}) is unstable. Then, it gets service $< (\lambda - \epsilon)t$ for arbitrarily small ϵ and large t ($>$ *some* T).

It is important to note that since at every time step, we choose a permutation matrix (we choose MSM and add extra edges arbitrarily to complete the permutation), we either choose the cross matching or parallel matching

If we treat VOQs Q_{12} & Q_{11} as a system, and observe these *unstable* queues over an extended period of time, then, the cumulative service rate in this system ($= 1$) exceeds the arrival rate, 2λ (< 1). Thus, at least one of these VOQs will go to zero. Similar arguments prove that Q_{21} cannot be unstable, if Q_{11} is unstable (as assumed).

Before proceeding further, it is important to note that $s_{11} = s_{22}$ (=

s *say*) and $s_{21} = s_{12}$ ($= (1 - s)$) over any interval of time because we either choose the *parallel* matching or the *cross* matching.

Using the fact that Q_{11} and Q_{22} will get the same service over any interval of time, Q_{22} will get the same service $< (\lambda - \epsilon)t$ over this interval t (for large enough t) which makes Q_{22} unstable. This contradicts the above stated lemma. Hence, we reach a contradiction.

An interesting open problem, that we are investigating, is to extend these arguments for `NxN` switches.

5 Wait-until-full

We looked at various variants of MSM algorithms including batched scheduling and alike. We investigated the stability of *wait-until-full* strategy under uniform traffic. This strategy is remotely connected to MSM type scheduling algorithms since this algorithm always schedules N packets (MSM at that instant) whenever any packets are switched. Its relation to MSM (or absolutely no relation to MSM) will become clear, when we formally state the algorithm.

Algorithm *Wait-until-full*

- If any VOQ is empty, serve no queues.
- If no VOQ is empty, pick a permutation uniformly at random across some sequence (or all) of permutations.

5.1 Stability of Wait-Until-Full

Observation 1 : Wait-until-full chooses permuatition *uar* when it services the VOQs, and thus all VOQs get the same service rate.

Theorem : *Wait-Until-Full* is stable under uniform, admissible Bernoulli i.i.d. arrival process.
Proof : Let $A_{ij}(s,t)$ and $D_{ij}(s,t)$ denote the cumulative arrivals and departures to the VOQ_{ij} in the time interval (s,t).
For Bernoulli arrivals, we know that
$A_{ij}(0,t) = \lambda \times t \pm O(\sqrt{t})$ (**1**), for large enough t, $\forall i,j$.

If any of the VOQs is unstable, then, it gets service $< (\lambda - \epsilon)t$ for arbitrarily small ϵ and large t ($>$ *some* T).

Using *Observation* 1, it follows that all VOQs get service $< (\lambda - \epsilon)t$

for some sufficiently large t. (**2**)

Furthermore, we know that,

$Q(0,t) = max_{0 \leq s \leq t}\{A(s,t) - D(s,t)\}$

which gives $Q_{ij}(0,t) = \epsilon t \pm O(\sqrt{t})\ \forall i,j$ using (1) and (2). (**3**)
Consider a time $t_1 = min_{t_i<t} t_i$ such that there is no VOQ that goes empty in the interval (t_1, t).
There must some VOQ (say VOQ_k) that is empty at t_1.

VOQ_k grows from 0 to $\epsilon t \pm O(\sqrt{t})$ in the interval (t_1, t) and by the definition of t_1, no VOQ goes empty in the time interval (t_1, t). By the property of Wait-Until-Full, all VOQs will be getting service at a rate of $\frac{1}{N}$ in the interval (t_1, t).
Therefore, VOQ_k cannot grow to $\epsilon t \pm O(\sqrt{t})$ for large t which contradicts (3).

Thus, *Wait-Until-Full* is stable under uniform, admissible Bernoulli i.i.d. arrival traffic.

6 Conclusions

Our work was motivated by simulation results that suggested that MSM and Wait-Until-Full are stable for uniform Bernoulli i.i.d. arrivals. We gave a counter-example showing that MSM is unstable for adversarial, uniform, admissible traffic if the MSM schedules are chosen adversarially. Although, we proved the stability of 2x2 switch under bernoulli i.i.d. traffic, we were not able to extend the arguments for general NxN switch which remains an open problem. Furthermore, we showed the stability of Wait-Until-Full algorithm under Bernoulli i.i.d. uniform traffic.

Acknowledgements

We would like to thank Sundar Iyer, Devavrat Shah and Isaac Keslassy for various discussions and providing pointers on this problem.

References

[1] Achieving 100% Throughput in an Input-Queued Switch, *N. McKeown*, *A. Mekkittikul*, *V. Anantharam* and *J. Walrand*, *IEEE Transactions on Communications*, Vol. 47, No. 8, August 1999.

[2] Maximum Size Matching and Input Queued Switches, *S. Iyer* and *N. McKeown*, *Proceedings of the* 40*th Annual Allerton Conference on Communication, Control and Computing*, 2002.

[3] Scheduling non-uniform traffic in a packet-switching system with small propagation delay, *T. Weller* and *B. Hajek*, *IEEE/ACM Transactions on Networking*, $5(6): 813-823$, 1997.

[4] Maximum Size Matching is Unstable for Any Packet Switch, *I. Keslassy*, *R. Zhang* and *N. McKeown*, Stanford HPNG Technical Report TR03-HPNG-030100.

[5] Randomized Scheduling Algorithms for High Aggregate Bandwidth Switches, *P. Giaccone*, *B. Prabhakar* and *D. Shah*

[6] Input Queued Switches in Heavy Traffic, *D. Wischik*, *J. Harrison*, *F. Kelly*, *S. Kumar*, *B. Prabhakar*, *D. Shah* and *R. Williams*, http://www.statslab.cam.ac.uk/ djw1005/Stats/Talks/switch.pdf