Performance Evaluation of an Optical Packet "Scheduling" Switch

Kyriakos Vlachos, *member IEEE*, Kyriaki Seklou and Emmanuel Varvarigos

Research Academic Computer Technology Institute, RA-CTI, University of Patras,
GR-26500, RIO, Patras, Greece, email: kvlachos@ceid.upatras.gr

Abstract— **In this paper a performance analysis of the packet scheduling switch is presented. The scheduling switch uses a series of feed forward delays interconnected with elementary optical switches. This series of programmable delay blocks constitute an optical buffer of depth T, whose purpose is to delay/re-arrange incoming packets that request the same outgoing link so as to resolve or reduce packet contention. Performance results have been obtained for random Bernoulli traffic, Pareto traffic, as well as for smooth traffic with an upper bound of inherent burstiness.**

Keywords-component; optical packet switching, burst traffic, packet scheduling, Pareto, optical networks

I. Introduction

Several innovative packet switch architectures have been proposed so far, including switches with re-circulating loops [1], the Staggering switch [2], the Switch with Large Optical Buffers (SLOB) [3], the Wavelength Routing Switch and the Broadcast (WRS) and Select Switch (BSS) [4]. However, work on new architectural concepts, node's performance, and intelligent control have lagged behind progress in transmission speeds.

Switches with recirculating loops were the first optical packet switches to address the high bandwidth and buffering issues [1]. This solution however increases switch block complexity, since for an *N*x*N* switch with *L* delay lines for buffering, instead of an *NxN*, an *(N+L)*x*(N+L)* space switch is required [5]. The Staggering switch was the first optical switch designed to truly emulate an output-buffered switch [2]. Although very promising and influential, this design exhibits unnecessary packet delays and unsatisfactory packet loss characteristics for bursty traffic. Reduction of the packet loss rate for bursty traffic is achieved by cascading many small output-buffered switches, consequently increasing costs, to arrive at a larger buffer depth. The SLOB is an example of such a design [3]. Renaud et al [4] detailed two WDM shared output-buffered packet switching architectures, namely the WRS and the BSS switch, that were developed through the ACTS KEOPS (Keys to Optical Packet Switching) project. The WRS is a two-stage switch that first buffers conflicting packets before routing them to their desired output, where a tunable wavelength converter is used to route packets to the appropriate delay line and output port respectively. The BSS architecture is one of the few proposed architectures that can easily provide multicasting. In addition, it can be used as the building block in a multi-stage switch, to allow for a modular growth, up to several hundreds of switch I/Os.

In this paper, we analyze the performance of the optical packet scheduling switch [6] for random Bernoulli traffic, Pareto traffic, as well as for constrained and unconstrained bursty traffic patterns. The scheduling switch uses a series of feed forward delays interconnected with optical switches to resolve internally packet contention, and it is guaranteed to be lossless when a certain smoothness property holds. In this paper we investigate the packet loss performance of the switch when this smoothness property does not hold. The analysis carried out shows that the scheduling switch exhibits very low packet loss ratio for random Bernoulli traffic, while it allows lossless communications for sessions that are subject to an upper bound of burstiness, hereinafter called (*n*,T) *smoothness* property. The remainder of the paper is organized as follows. Section II presents the scheduling switch architecture and the (*n*,T) smoothness property. Section III presents performance evaluation results, while Section IV concludes the paper and proposes some future work.

II. Switch Architectures and Smoothed Traffic Model

The scheduling switch has been designed to provide lossless communication for sessions that have a certain burstiness property or can be transformed to sessions with such a property, tolerating the corresponding delay. It consists of a scheduling unit with *k* input/output ports, and a *k*x*k* non-blocking space switch, as shown in Figure 1. Each branch delays the incoming packets, assigning packets to outgoing slots resolving contention and maintaining packet ordering for the same outgoing link. The problem of scheduling packets through a branch of delay blocks to avoid collisions is a problem of routing a permutation between inputs and outputs in the equivalent Benes network, where non-overlapping paths in the network correspond to collision-free transmission through the delay blocks [6].

Each delay branch consist of 2*m*-1 delay blocks, where *m = logT. T* is assumed to be a power of 2 and corresponds to the maximum number of sequential packets from all incoming links that request the same output and can be served with no contention. The i^{th} block consists of a three-state (or two 2x2) optical switch and three fiber delay paths, corresponding to delays equal to 0, 2^i and 2^{i+1} packet slots. To ensure that the packets in the incoming frame can be assigned to any slot in

This work was supported in part by EC via the e-Photon/ONe project

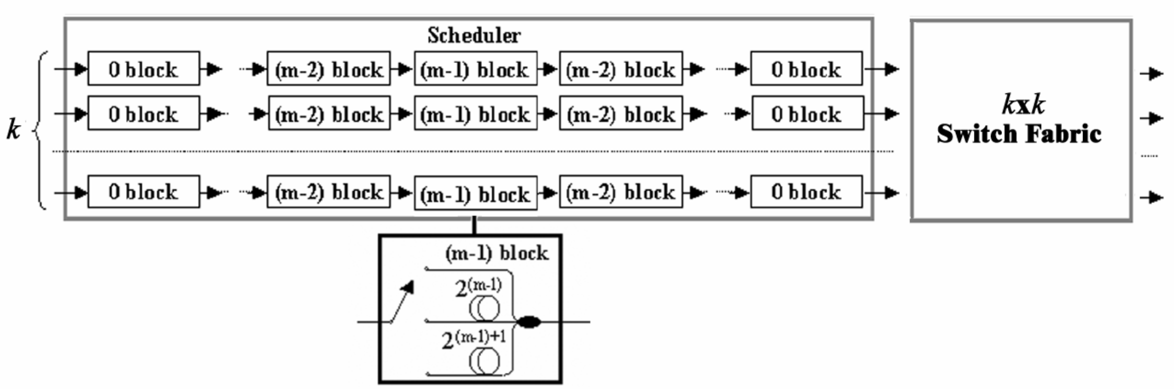

Figure 1: The Scheduling Switch architecture, consisting of the Scheduler (k inputs) and a kxk space switch. The Scheduler comprises of k branches, each with 2log2T-1 delay blocks. The i-th delay block consists of one three-state switch and three delay lines of length 0, 2i and 2i+1 packet slots.

the outgoing frame, the latter must start at least (3T)/2-2 after the incoming frame begins. Such an input queuing scheme can be viewed as implementing an optical packet buffer of depth *T* [7]. One of the major advantages of the scheduling packet switch is its modular buffering scheme design that can be easily expanded to accommodate more burstiness in the traffic (in a way similar to the way electronic buffers can be expanded in a conventional electronic switch). The cost of the switch, when measured in terms of the elementary switching elements it requires, grows only logarithmically with *T*.

A corresponding traffic model that can guarantee lossless communication must be based on the aforementioned buffering scheme. To this end, we assume that the time axis on a link is divided into packet slots of equal length and every *T* slots are virtually grouped to form a frame. This concept is illustrated in Figure 2. Packets are grouped in *T*-size frames before entering the switch, while frame integrity is maintained at the output as well. A session is said to have the *(n,T)- smoothness* property at a node if at most *n* packets of the session arrive at that node during a frame of size *T*. A session can easily be transformed to have the *(n,T) – smoothness* property at the ingress point of the network while this property can be preserved throughout a network consisting of scheduling switches, due to frame integrity maintenance.

We let n_{ij} be the number of packets that arrive during a frame over incoming link *i* and have to be transmitted on link *j*, and *k* the number of incoming (and outgoing) links of a node. If the connection and flow control protocols used guarantee that the number of packets which require the same outgoing link *j* in a frame is less than or equal to the frame size *T*, i.e.,

$$\sum_{i=1}^{k} n_{i,j} \leq T \qquad \text{for all } j \in \{1,2,..k\}. \qquad (1)$$

then all of the incoming packets can be assigned slots in the required outgoing links so that no packets are dropped. Both wait-for-reservation and tell-and-go protocols can be used to ensure that this property is met. The frame size *T* is an important parameter and can be viewed as a measure of the *traffic burstiness* allowed. The larger *T* is, the less constrained (more bursty) is the incoming traffic allowed to be, and the larger is the flexibility *–granularity–* in assigning rates to sessions. For example if each link in a network has capacity *C* and a session has the *(n,T)*-smoothness property, then this session will have an average rate of at most *nC/T*, implying that capacity can only be allocated in discrete multiples of *C/T*. It is important to note that this is not circuit switched data, but instead, this is packet switching with built-in flow control to ensure lossless transmission. Packets from a particular source do not arrive in the same slot and the number of packets that arrive per frame is not constant, but is bounded by *n*.

III. Scheduling Switch Performance

In this section, we present results on the packet loss performance of the scheduling switch for random Bernoulli traffic, for bursty traffic based on a truncated Pareto packet distribution model, as well as for *(n,*$T_{traffic}$*)* smooth traffic with a parameter $T_{traffic}$ different than the corresponding parameter *T* used in the switch design. For the aforementioned case studies, the scheduling switch is considered as a packet switch with *k* input buffers, each with *T* available packet slots.

Regarding our first traffic scenario, we assume that packets arrive at each switch input according to a binomial process, and their destinations are uniformly distributed over all output ports. In reality, traffic is much more bursty than that; more specifically, Internet and voice traffic have been shown to be better modeled by Pareto and exponentially distributed statistics, respectively. The model of independent Bernoulli processes is the simplest model that can be considered, resulting in a tractable analysis, while still yielding an

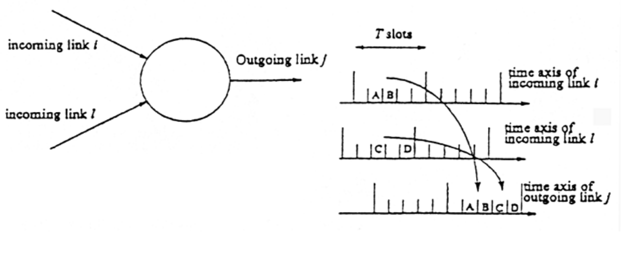
incoming link i
incoming link l
Outgoing link j
T slots
A B
time axis of incoming link i
C D
time axis of incoming link l
A B C D
time axis of outgoing link j

Figure 2: Incoming and outgoing frames at a node. Packets arriving in a particular frame of an incoming link that want to use the same outgoing link are sent over the same frame of the outgoing link.

appreciation of switch performance. More bursty models for the traffic will be considered later.

Assuming that packets arrive independently at each incoming slot with probability p, then the probability of having i packets arrivals during the kT slots of the k incoming frames requesting the same output j, $j = 1\ldots k$, and assuming uniformly distributed destinations is:

$$P[X=i]=\binom{kT}{i}\cdot\left(\frac{p}{k}\right)^{i}\cdot\left(1-\frac{p}{k}\right)^{kT-i} \qquad (2)$$

The packet loss ratio can then be easily calculated as:

$$PLR=\frac{\sum_{i=T}^{kT}P[X=i]\cdot(i-T)}{p\cdot T} =\frac{\sum_{i=T}^{kT}\left[\binom{kT}{i}\cdot\left(\frac{p}{k}\right)^{i}\cdot\left(1-\frac{p}{k}\right)^{kT-i}\cdot(i-T)\right]}{p\cdot T} =\frac{\sum_{i=T}^{kT}\left[\left(\frac{kT!}{i!\cdot(kT-i)!}\right)\cdot\left(\frac{p}{k}\right)^{i}\cdot\left(1-\frac{p}{k}\right)^{kT-i}\cdot(i-T)\right]}{p\cdot T} \qquad (3)$$

In obtaining this equation, we used the fact that the switch maintains frame integrity and that if more than T packets arrive during the k incoming frames heading for output j, the excess packets are lost. Figure 3(a) and (b) show the packet loss ratio as a function of link utilization for a scheduling switch with k=2 and k=4 input/output ports, while T is varied from 2 to 1024. From Figure 3, it can be seen that the packet loss ratio is very low for values of T higher than 32 and $p < 0.8$. The parameter T can also be viewed as a measure of the buffer size available per input. The buffering that can be accomplished with the scheduling switch is considerably higher than the buffering that can be accomplished with other switch architectures that use e.g. fiber delay lines [4],[5], for the same implementation cost, because of the logarithmic dependence of the complexity of the scheduling switch on T.

Although the scheduling switch is quite complex requiring the integration of numerous elementary switches, however improvement in its architecture with state-of the art, all-optical technologies can downsize its cost and complexity [8]. We believe that the scheduling switch architecture, where buffering is accomplished using a logarithmic number of elementary 2x2 optical switches, offers a feasible way to design modular, high capacity all-optical buffers that can take advantage of statistical multiplexing. This is more clear in Figure 4(a) and (b) that display, for a given utilization factor, the way the packet loss ratio varies with T. From these figures, it can be seen that the loss ratio drops very fast as T increases. Even with 100% utilization (p=1), PLR can be very small when T is of the order of 1024 or larger. Increasing T by a factor of 2 requires the addition of two delay blocks –and thus four 2x2 switches– per switch port. The modularity of the scheduling architecture design allows us to increase the throughout or be able to accommodate more bursty traffic for a given PLR with only a moderate increase in the complexity and the cost of the switch.

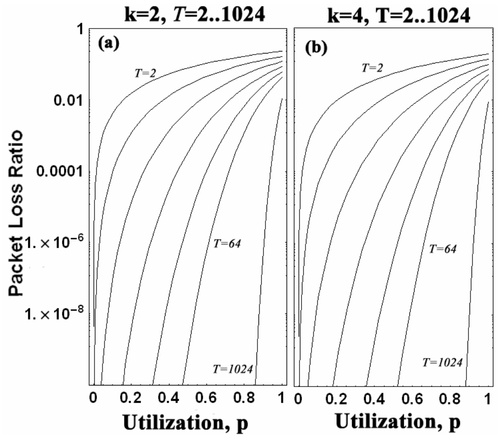

Figure 3: Packet loss ratio for (a) k=2 and (b) k=4 for binomial packet traffic and uniformly distributed destinations

In our second case study, we assume that the incoming traffic obeys the *(n,T)* smoothness property, but with a parameter T higher than the one the scheduling switch has been designed to tolerate. More specifically, we assume that the parameter T of the traffic, denoted by $T_{traffic}$, is an integer multiple of the corresponding parameter T of the switch, denoted by T_{switch}. The ratio $\mathrm{T}_{traffic}$ / T_{switch} can be treated as an index of the traffic burstiness in a Scheduling-switch based optical network. The link utilization is assumed to be equal to p, meaning that the total number of packets arriving over all inputs in an incoming frame of size $T_{traffic}$ that request the same outgoing link j is:

$$\sum_{i=1}^{k} n_{i,j} = pT_{traffic} \qquad \text{for all outputs } j.$$

The position of the packets of an $(n,T_{traffic})$ smooth session within an incoming frame is assumed to be distributed uniformly among all slots of the incoming frame of size $T_{traffic}$. The probability of having i packets within the T_{switch} first slots

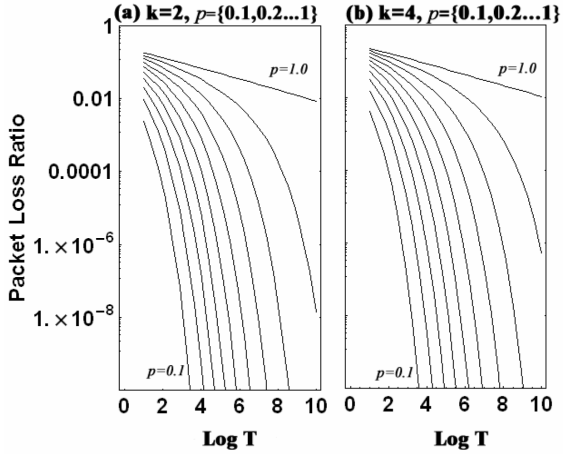

Figure 4: Packet loss ratio versus T for (a) k=2 and (b) k=4 and for a utilization p = {0.1, 0.2,…1}. For p=1, packet loss ratio is $9\cdot10^{-3}$ and $11\cdot10^{-3}$, when T=2^{10} for k=2 and k=4 respectively.

of the k incoming frames is:

$$P_i = \frac{\binom{kT_{switch}}{i} \cdot \binom{kT_{traffic} - kT_{switch}}{pT_{traffic} - i}}{\binom{kT_{traffic}}{pT_{traffic}}} \tag{4}$$

From Eq. (4) we can easily derive the corresponding packet loss ratio:

$$PLR = \frac{\sum_{i=T_{switch}}^{pT_{traffic}} \left[\frac{\binom{kT_{switch}}{i} \cdot \binom{kT_{traffic} - kT_{switch}}{pT_1 - i}}{\binom{kT_{traffic}}{pT_{traffic}}} \cdot (i - T_{switch}) \right]}{pT_{traffic}} \tag{5}$$

In the above packet loss ratio calculation, we have used the fact that the $pT_{traffic}$ packets that arrive per incoming frame and request output j are evenly distributed within the frame of size $T_{traffic}$, and can arrive in any of the $\binom{kT_{traffic}}{pT_{traffic}}$ possible combinations. If more than T_{switch} packets arrive during the first kT_{switch} incoming slots, the excess packets will be dropped. Figure 5a and b display the loss probability curves for k =2 and k = 4. In these figures, fixed values of T_{switch} equal to 2 and 16 have been considered, while $T_{traffic}$ is varied to integer multiples of the aforementioned T_{switch} values. It is worth noting that Eq. (5) is valid only for $pT_{traffic} > T_{switch}$, while for $pT_{traffic} = T_{switch}$ or $T_{traffic} = T_{switch}$, the packet loss ratio is zero for any utilization factor p. From both figures it can be seen that when $T_{traffic}/T_{switch}$ increases (beyond 2), the packet loss ratio decreases. This is primarily due to the burstiness averaging as a result of the numerous possible packet distributions within a $T_{traffic}$ frame.

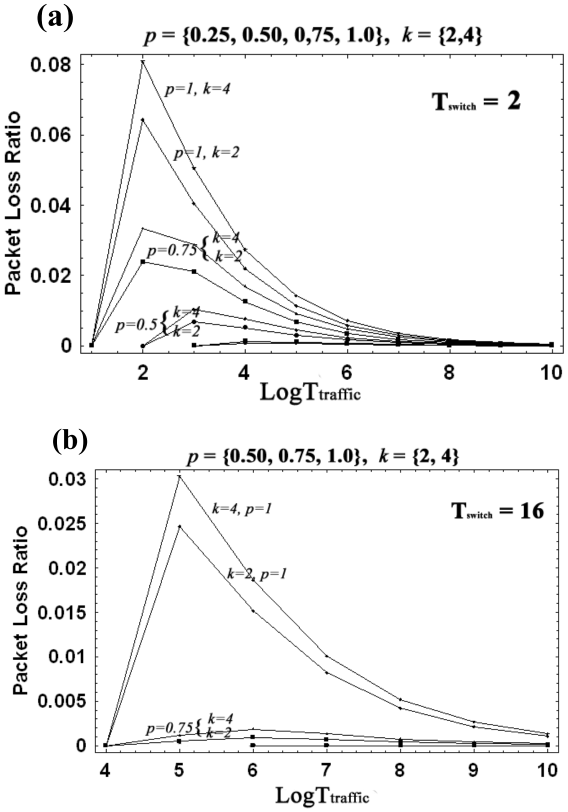

Figure 5: Packet loss ratio for (a) T_{switch}=2 and (b) T_{switch}= 16, versus the $T_{traffic}$ / T_{switch} ratio for a k=2 and k=4 scheduling switch and a utilization p = {0.25, 0.5, 0.75, 1}. $T_{traffic}$ is varied from $2T_{switch}$ to 2^{10}.

In the third traffic model we consider, we investigate the performance of the scheduling switch under a heavy-tailed truncated Pareto distribution, which is considered by many researchers as a good model for burst traffic in real networks. In our model, packets arrive in bursts (ON periods), which are separated by idle periods (OFF periods). To generate a Pareto-distributed sequence of ON periods, one can generate a Pareto-distributed sequence of burst (packet train) sizes, followed by Pareto-distributed idle times. The minimum burst size is 1 corresponding to a single packet arrival. The formula to generate a Pareto distribution is: $X_{PARETO} = \frac{b}{x^{1/a}}$, where x is a uniformly distributed value in the range (0, 1], b is the minimum non-zero value of X_{PARETO}, denoted by b_{on} and b_{off} for the packet train and idle period respectively and a the tail index or shape parameter of the Pareto distribution. However computer simulations using the above formula generate a truncated Pareto distribution, because of the discreet x value. On the contrary, any true Pareto distribution of sufficiently large length will have values that exceed the range generated by computer simulations. To this end, since the mean size of the truncated Pareto distribution differs from the mean size of a true one, the question that rises is which is the minimum possible idle period so that on average the truncated Pareto distribution yields a link utilization factor of p. In order to define b_{off}, we have assumed, first fixed length packets and secondly idle periods to be equal to an integer multiple of a single –fixed length- packet. To this end, by expressing the utilization factor p as the mean size of ON period over mean size of ON and OFF periods:

$$p = \frac{\overline{ON}_{period}}{\overline{ON}_{period} + \overline{OFF}_{period}} \tag{6}$$

, calculating the mean value of the truncated Pareto distribution, which does not exceed the value $X_{Pareto}^{\max} = \frac{b}{x_{\min}^{1/a}}$, as shown below:

$$E(x) = \int_{b}^{X_{Pareto}^{\max}} x f(x) dx = \int_{b}^{X_{Pareto}^{\max}} x \frac{ab^{a}}{x^{a+1}} dx = \frac{ab}{a-1}\left[1 - x_{\min}^{\frac{a-1}{a}}\right] \qquad (7)$$

and substituting eq. (7) to (6), the minimum idle period as a function of link utilization can be derived [9]:

$$b_{off} = \frac{\frac{a_{off}-1}{a_{off}}}{\frac{a_{on}-1}{a_{on}}} \cdot \frac{1 - x_{\min}^{\frac{a_{on}-1}{a_{on}}}}{1 - x_{\min}^{\frac{a_{off}-1}{a_{off}}}} \cdot \left(\frac{1}{p} - 1\right) \qquad (8)$$

In the above equations, *f(x)* is the probability density function of the Pareto distribution, $x_{\min}$ is the smallest non-zero value of x that is uniformly distributed in (0,1], and α_{on}, α_{off} are the tail indices for the packet train and idle period size respectively. It is worth noticing here that our intention was to generate traffic load being very close to the specified load with all combinations of a_{on}, and a_{off}.

After defining b_{off}, we performed computer simulations for a $k = 2$ and $k = 4$ scheduling switch with $\alpha_{on} = 1.7$, $\alpha_{off} = 1{,}2$ and $X_{\min}=10^{-4}$. Figure 6 show the corresponding loss ratio results for T ϵ [2…64]. Again packet destinations were evenly distributed. In our simulation, we have selected α_{on} to be larger that α_{off}, since in real traffic, the probability of having extremely large OFF period is higher then the probability of having extremely large ON periods. From Figure 6, it can be seen that the scheduling switch loss ratios, in the case of Pareto distribution differ significantly from the ones shown in Figure 3 for random Bernoulli traffic. This is more evident for small T values and is attributed to the bursty nature of the Pareto distribution. More specifically, since the mean size of the ON periods, ~2.4, is close to T then during an ON period, all slots of the frame are filled, independent of the resulting workload. This results in increased packet loss ratios and especially in the case of T = 2, it can be noted that loss varies slightly for all *p*. This is due to the fact that T is smaller than the mean value of ON periods. Nevertheless, as T increases, packet loss ratio drops fast and for T values higher than 32, an acceptable loss ratio of 10^{-6} for $p < 0.6$ is obtained.

IV. Conclusions

In this paper, we have analyzed the performance of the scheduling switch under a variety of models for the incoming traffic. The scheduling switch is guaranteed to be lossless when the incoming traffic has the so-called (*n*,T) smoothness property. We have evaluated the switch performance, when this property does not hold for three different traffic models namely, a Bernoulli model, a truncated Pareto model, and a model where traffic is constrained but with a smoothness parameter different than the one used in the switch design. In a future communication, the delay impairments at the network ingress point due to the (n,T) smoothness property enforcement will be investigated and a suitable edge router architecture will be proposed.

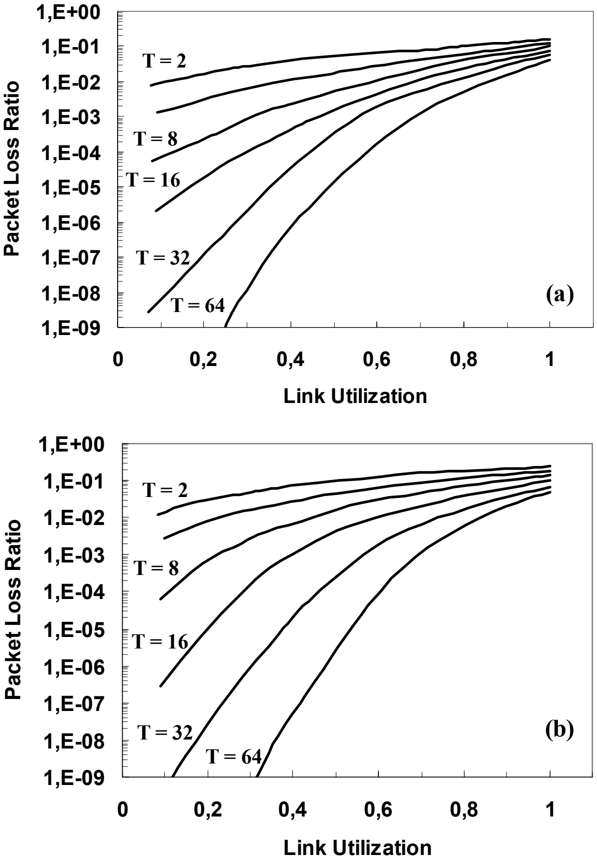

Figure 6: Packet loss ratio for (a) k=2 and (b) k=4 versus link utilization for T ϵ [2…64]. Packet arrivals and idle periods follow a truncated Pareto distribution with a tail index of 1.7 and 1.2 respectively.

References

[1] A. Huang and S. Knauer, "Starlight: A wideband digital switch," in Proc. IEEE Global Telecommun. Conf. (GLOBECOM'84) vol. 1, pp. 121–5, Nov. 1984

[2] Z. Haas, "The "Staggering switch: An electronically controlled optical packet switch," J. Lightwave Technol., vol. 11, pp. 925–36, 1993.

[3] D. Hunter et al., "SLOB: A switch with large optical buffers for packet switching," J. Lightwave Technol., Vol. 16, pp. 1725–1736, 1998.

[4] M. Renaud et al. "Network and system concepts for optical packet switching," IEEE Commun. Mag., vol. 35, pp. 96–102, Apr. 1997.

[5] K. Vlachos, I.T. Monroy, A.M.J. Koonen, C. Peucheret and P. Jeppesen "STOLAS: Switching Technologies for Optical Label Signals", IEEE Communications Magazine, vol. 41, no. 11, pp. 43-49, November 2003.

[6] J.P. Lang, E.A. Varvarigos and D.J. Blumenthal, "The lambda – scheduler: A multiwavelength scheduling switch", IEEE J. of Lightw. Techn., vol.18, no.8, pp. 1049-1063, 2000.

[7] I. Chlamtac et al. "CORD: Contention Resolution by Delay Lines", IEEE Journal on Selected Areas in Communications, vol. 14, no. 5, pp. 1014 –1029, June 1996.

[8] G. Theophilopoulos et al. "An alternative implementation technique for the scheduling switch architecture", accepted for publication in IEEE J. of Lightw. Techn.

[9] G. Kramer, "On generating self-similar traffic using pseudo-Pareto distribution" Technical brief, Department of Computer Science, University of California, Davis. <http://wwwcsif.cs.ucdavis.edu/~kramer /papers/self_sim.pdf>