Towards a Bufferless Optical Internet

Eric W. M. Wong, *Senior Member, IEEE*, Lachlan L. H. Andrew, *Senior Member, IEEE*, Tony Cui,
Bill Moran, *Member, IEEE*, Andrew Zalesky, *Member, IEEE*, Rodney S. Tucker, *Fellow, IEEE/OSA*, and
Moshe Zukerman, *Fellow, IEEE*

Abstract— This paper investigates the relationship between buffer size and long-term average TCP performance in dense wavelength division multiplexing (DWDM) networks. By investigating TCP NewReno, we demonstrate that buffer requirements are related to the number of wavelength channels at a bottleneck. With sufficient wavelengths, high throughput can be obtained with a buffer of one packet per channel; furthermore, there may be situations where an entirely bufferless optical packet switching (OPS) will become feasible.

For this study, we develop new evaluation tools. First, we propose a method based on a two-part analytical model, with a new "open loop" component which approximates packet discarding in a bottleneck DWDM switch, and a "closed loop" fixed point which reflects the impact of TCP. This analytical method provides accurate and scalable approximations of throughput and packet loss rate that can be used as part of a tool for DWDM network and switch design. Second, we propose an extrapolation technique to allow simulation of TCP over long ultra-high bit rate links, avoiding the intractable processing and memory requirements of direct simulation. This extrapolation technique enables us to validate the analytical model for arbitrarily high bit rate scenarios.

Index Terms— TCP, DWDM network, bufferless optical packet switching (OPS).

I. Introduction

In the past, it had been widely believed that Internet switches need large buffers to achieve high throughput. A common rule-of-thumb [1], [2] states that a switch needs a bitrate-delay product (BDP) of buffering, $B = C_0 \times \overline{RTT}$, in order to fully utilize bottleneck links. Here, B is the size of the buffer, C_O is the capacity of the bottleneck link, and $\overline{RTT}$ is the average round trip time of the TCP flows running in the bottleneck link (where "round trip time" is the time between when a packet is sent and when its acknowledgment is received, when all queues are empty). Internet design using

large buffers results in many queued and hence delayed packets in the network, and these packets delay other packets. It is therefore advantageous to consider designs that reduce queueing delay while still maintaining high network utilization.

Appenzeller *et al.* [3] proposed to use the rule $B = C_0 \times \overline{RTT}/\sqrt{N}$ instead, where N is the number of simultaneous TCP flows. As argued in [4], [5], there is interdependence between small buffers in routers and network stability. Enach-escu *et al.* [6] then suggested that the buffer size need not be larger than $O(\log(W))$, where W is the maximum window size of the TCP flow control used by the operating system of the receiver. Based on this rule, Beheshti *et al.* [7] suggested that optical packet switches may only need 10-20 packet buffers, at the cost of a certain reduction in utilization. None of these studies has considered the effect of wavelength division multiplexing (WDM) or dense WDM (DWDM) [8], an important feature of present and future networks, in which there are multiple wavelength channels on each core trunk.

Eramo *et al.* [9], [10] showed that switches with highly symmetric load need only a small number of wavelength converters to achieve a low packet loss probability. Wong and Zukerman [11] proposed a queueing model for DWDM switches and demonstrated that only small, but possibly non-zero, buffers are needed if many wavelength channels per trunk and full wavelength conversion are available.

These early publications indicate limited need for buffers for TCP traffic and in DWDM switches. However, the interaction between **these two** has not been studied previously. In this paper we consider a "closed-loop" model of a scenario that includes both TCP and core trunks with multiple wavelength channels and we provide a scalable and accurate analytical method for the evaluation of TCP throughput and packet loss rate for DWDM networks. This method enables practical conclusions on buffer sizing to be drawn and can lead to a useful tool for network and switch design.

The small buffer requirements predicted by [6], [7] are partially due to the limit imposed on TCP windows by today's small receiver buffers. In the present paper, we place no limit on the maximum window size. We assume that the TCP sender and receiver buffers are large enough that TCP's congestion window determines the transmission rate. This is a conservative **assumption** and is likely to be necessary for future networks.

We consider the network model shown in Figure 1. TCP sources transmit packets via access links to edge routers (ERs) which have large electronic buffers that have negligible probability of overflowing. **The TCP sources are assumed to be greedy, i.e., they always have data to transmit.** From the ERs, the packets are transmitted to a symmetric core DWDM
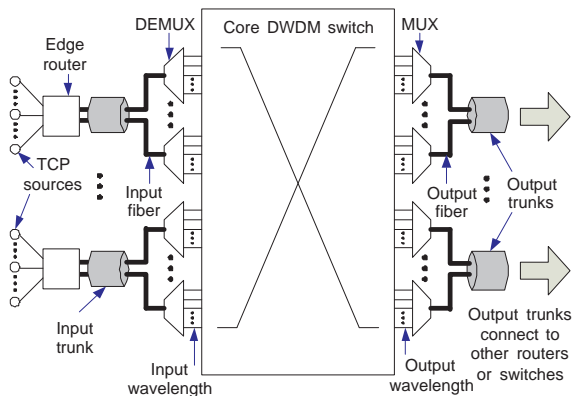
Fig. 1. Network topology of an optical network with a bottleneck symmetric switch

switch from which they are forwarded to other switches. Note that our model is also applicable to topologies that do not involve ERs, but instead allow packets from one or more sources to be stored in a router on the customer premises and transmitted directly to the network.

The analysis presented in this paper applies to electronic packet switches and to optical packet switches, with buffers of equal (possibly zero) size at each output port. However, the demonstration of zero buffering requirement is particularly relevant to optical switching.

This paper also covers the case of hybrid (optical/electronic) switching, where, for instance, buffering and wavelength conversion are performed electronically but incoming packets on a given wavelength may cut through optically if the same wavelength is available at the output [14], [15].

We will use the following terminology and notation. A connection between two nodes is called a trunk. A trunk consists of independent *links* (also called *wavelength channels*). Every link is connected to the switch at a *port*. Each input link terminates on an input port of the switch and each output link originates at an output port of switch. Note that we consider the typical case where the number of TCP sources is much larger than the total number of input links on all input trunks. Moreover, since there are multiple trunks, the total number of input links is larger than the number of output links on any single output trunk. We **consider a non-blocking switch** with output buffering and full wavelength conversion.

Note that true electronic output buffering is not feasible at high speeds, but it can be exactly emulated at high speed using virtual output queues [16]. Moreover, output buffering can be achieved in optical switches using multiple-input single-output buffers consisting of a combination of space switching and multiple FIFO delay lines [12], [13]. For a switch that is not fully non-blocking, the overall throughput achieved will be lower due to the additional packet discards.

We consider a switch with L input and output trunks, where $L > 2$. Each trunk consists of K links. Consider a subsystem consisting of the K links/ports of an output trunk and all the packets arriving at the $M = (L-1)K$ input ports that aim to access that output trunk. For any output trunk, to evaluate

the buffer size required, we assume for simplicity that this trunk is the only bottleneck for all the sources sharing it. Accordingly, we consider only a single subsystem associated with one output trunk and all the sources that transmit packets through it, and all the ERs, links and ports that forward the packets to it.

As in previous work [6], [7], we consider switches in which each individual connection carries a small fraction of the total capacity, allowing us to neglect the "saw-tooth" changes in TCP rate due to additive increase and multiplicative decrease. This is justified by the fact that most traffic in core switches must traverse access links with rates orders of magnitude smaller than those of the optical links rather than by assuming an artificial limit on the receiver's window as done previously. Specific networks, such as dedicated high-speed scientific networks [20], may require different buffer sizing rules.

The contribution of this paper is fourfold, with the development of new performance evaluation tools as well as the buffer-sizing study.

The first contribution, in Sections II and III, is a scalable "open loop" approximation model for the packet loss probability as a function of traffic load. Here the traffic load is defined as the rate [packets/s] times the mean packet size [bits]. We demonstrate that our approximation is accurate in a wide range of scenarios. Although the open loop model is a component of our closed loop model, it has a value on its own as it is applicable to dimensioning networks that carry a significant amount of non-TCP traffic, which are expected to be prevalent [21].

The second contribution, in Section II-B, is a closed-loop analytical model to estimate bottleneck throughput. This model includes the open-loop model and also the feedback provided by TCP, whose throughput is determined by the packet loss probability [22], [23]. A fixed-point solution of this closed-loop model can be calculated by a binary search algorithm. By comparing it with ns2 [24] simulation results, we show that this analytical model provides an accurate approximation to the throughput.

Third, in Section IV we provide an extrapolation method to estimate the performance of high bit rate optical networks. Having a high bit rate means that simulations require a large amount of memory to store all the packets which are "in flight". Moreover, current CPUs require a very long time to process the large number of packets. Moore's law [25] cannot alleviate this plight because bit rates are scaling up as fast as the volume of memory and speed of CPUs [26]. The extrapolation method simulates a network with the same topology but lower bit rates and fewer flows, and then scales the results up to the desired bit rate [27].

Finally, Section V contains the main numerical study. We show that increasing the number of wavelengths dramatically reduces the buffering requirements, and that previous rules-of-thumb which disregard this effect will be overly conservative. The impact of **TCP Pacing [28], [29]** is also investigated, and we demonstrate that it will typically not have a major effect on the throughput of high-bandwidth core switches.
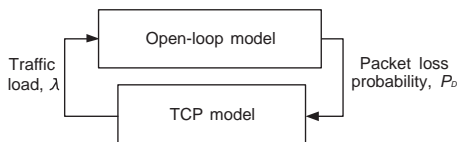
Fig. 2. Interaction between TCP protocol and the bottleneck DWDM switch

II. MODEL

Consider a DWDM switch carrying TCP traffic. The throughput and packet discard probability of this system will be studied by decoupling the closed-loop TCP-over-DWDM system and anlyzing two interconnected models: one is a novel "open-loop" queueing model and the other is a standard TCP model. This is illustrated in Figure 2.

After introducing the notation, the rest of this section will describe the TCP model. The open-loop queueing model is more complex, and will be described in the following section.

A. Notation

Packets arrive at the DWDM switch through any of its M input ports. For each input port, let $1/\lambda$ [seconds] be the mean idle time between two consecutive packets, measured from the end of one packet to the start of the next.

The *packet time* is the time it takes for an entire packet to be received by an input port. Let $1/\mu$ [seconds] be the mean packet time. An input port is said to be *busy* during the period it receives a packet.

The switch has a buffer consisting of B buffer spaces. Each buffer space can store a single packet indefinitely, and is available to store a new packet if either it is not currently storing a packet, or the packet it is storing **has been completely received and** is currently being output. An incoming packet will be discarded if there is neither an idle output port nor an available buffer space. The packet loss probability, P_D, estimates the proportion of packets which are discarded by the input ports.

The average rate of TCP flow is denoted by r [packet/second], and RTT [seconds] is the round trip time.

B. TCP model

Mathis *et al.* [22] proposed that the rate of a TCP flow (in packets/second) as a function of the packet loss probability is

$$r = \frac{\sqrt{1.5/P_D}}{RTT} \approx \frac{1.22/\sqrt{P_D}}{RTT} \qquad (1)$$

if the flow is dominated by TCP's "congestion avoidance" phase [30]. This equation applies to a family of TCP protocols, called by us "Reno", including Reno [30], NewReno [31] and SACK [32], which are all essentially the same at the level of abstraction we consider. More accurate expressions have been developed (see for example [23]). However if each flow is a small fraction of the total throughput, bursts are smoothed by the access links, and the buffer is sufficiently small for queueing delay at this switch to be negligible, then (1) is accurate enough for our purposes, as demonstrated by the simulations in Section V.

If all the TCP flows have the same long-term average packet loss probability P_D, the effective aggregate rate of all the TCP flows is

$$R_{agg} = \frac{N\sqrt{1.5}/\sqrt{P_D}}{\overline{RTT}_H} \qquad (2)$$

where R_{agg} [packets/sec.] is the aggregate rate of TCP flows, N is the total number of TCP flows and $\overline{RTT}_H$ is the weighted harmonic mean round trip time of these N TCP flows, namely $\overline{RTT}_H = R_{agg}/\sum_i(r_i/RTT_i)$, where r_i and RTT_i are the rate and round trip time of the ith flow, and the sum is taken over all flows.

The total packet arrival rate to the open-loop model is

$$\Lambda = \frac{M}{\lambda^{-1} + \mu^{-1}}, \qquad (3)$$

where the μ^{-1} reflects the reduction in load caused by each arrival, whether the packet is buffered, discarded or sent immediately, and Λ is equal to the aggregate packet arrival rate that all the TCP sources attempt to transmit, i.e. $\Lambda = R_{agg}$. From (2) and (3), the relationship between λ and P_D is $\lambda = T(P_D)$ where

$$T(p) = \frac{N\mu\sqrt{1.5}}{M\mu\overline{RTT}_H\sqrt{p} - N\sqrt{1.5}}. \qquad (4)$$

When all RTTs are equal, the total bitrate-delay product is $M\mu RTT$. **The model (1) from [22] does not consider timeouts, and hence only applies when flows have a window of at least four packets, to allow packet discards to be detected by three duplicate acknowledgements. The proposed model should thus only be used when $M\mu\overline{RTT}_H > 4N$.** Actually, for very low RTTs (4) is nonsensical; notice that if $M\mu\overline{RTT}_H < \sqrt{1.5}N$, the arrival rate $T(p)$ becomes negative.

Note that the model (2) applies to TCP Reno. Most new TCP variants achieve higher throughput for a given value of P_D, and so can tolerate smaller buffers. This is particularly true of TCP variants specifically designed for small-buffer networks [33]. Designing buffers for Reno is therefore a conservative approach.

C. Closed loop model

The foregoing TCP model finds λ in terms of its input: P_D and the system parameters RTT, N, M and μ. The open loop queueing model to be derived in Section III will find P_D in given its input: λ and the system parameters N, M, μ, B and K. Together, these two form a set of fixed-point equations, which we call the closed loop model. This set of fixed-point equations allows λ, P_D and, most importantly, the throughputs r and R_{agg} to be calculated.

III. OPEN LOOP MODEL — EVALUATION OF PACKET LOSS PROBABILITY

As discussed, the open loop model takes as input the traffic intensity λ along with parameters μ, M, K and B, and estimates the packet loss probability P_D as output. This section provides a detailed description of the open loop model and the method used to evaluate the packet loss probability. It

also provides information on previous related publications and several alternative models.

The model makes a mild symmetry assumption: For a given output port, there is a constant such that the rate of arrivals at each input port of packets destined for this output is either zero or that constant. Note that this does not require that the load on each output port be equal (as assumed in work such as [9], [10]), that the number of TCP flows using each input port be equal, that each TCP flow obtain equal rate, or that all pairs of input/output ports be used.

The model is based on a Markov chain, and implicitly assumes that both idle time and packet transmission time are exponentially distributed. **The former is justifiable in our case where we consider a large number of greedy flows.**[1] The latter, which was made for tractability, is not as limiting as one might expect; for instance, certain bufferless systems are entirely insensitive to the packet length distribution [34]. **However, in the case of large buffers, and considering realistic Internet traffic scenarios [35], [36], [37], [38], larger losses may occur than those predicted by our model based on these assumptions.**

In principle, a multidimensional continuous-time Markov chain can be defined to yield an exact solution of the above described model, but because the size of the state space of the Markov chain model is exponential in the model parameters, such a solution will be computationally prohibitive. The state space explosion results from the many possible interactions between input ports, buffers and output ports. Notice that an input port is busy when it is 1) switching a packet to an output port, 2) loading a packet into a buffer space, or 3) discarding a packet. A prohibitive number of states are required to record the states of each of the M input ports and the K output ports, and for each logical buffer position to record the identity of the input port from which the packet is written to it, if any. Therefore, we aim here to develop a model that will lead to an accurate yet scalable approximation. Given that M and K increase with technology advancements, our model should be of a single dimension to achieve scalability.

A. Prior work

The well-known Engset model [39], [40] which was originally developed for telephony, can be used to evaluate packet loss probability. However, it does not consider two effects, important for the present case: it assumes, firstly, that there is no buffer in the switch and, secondly, that an entire packet arrives instantaneously. This means that the Engset model allows other packets to arrive at an input port while a packet is being discarded, thus neglecting the reduction in traffic load due to discarded packets. Cohen [41] introduced a generalized version of the Engset model in which there is an exponentially distributed delay with mean $1/D$ after which a call is blocked during which the source cannot make another call. Setting $D = \mu$, this model is sufficiently versatile to include the

[1]Although TCP traffic is known to be correlated within a TCP session and among TCP sessions, the simulations in Section V-B demonstrate that the exponential approximation accurately predicts the queueing performance, at least when the link is highly multiplexed and the load is not time varying (e.g., a fixed number of greedy flows).

above reduction in load. However, Cohen's generalized Engset model (GEM) is still relevant only to a bufferless switch and it does not lend itself to a simple solution [41], [42], [43]. Only recently, by using advanced matrix methods [44], has an exact numerical solution for the blocking probability been achieved for the bufferless GEM for practical cases of hundreds of links per trunk.

An attempt to introduce a buffer to the GEM was made in [11], henceforth referred to as the WZ model, which is limited by the assumption that no packet can arrive on an incoming wavelength channel if a packet from that channel is already in the switch. This limiting assumption is relaxed in the model we introduce next which we call *Packet Engset with Buffer* (PEB). Afterwards we describe two special cases of the PEB, and in Section V, we provide numerical results over a wide range of parameter values that compare between the various modeling approaches.

B. Model: Packet Engset with Buffer

The PEB model is a single dimension ("birth-and-death") Markov chain. The state i of the Markov chain is the number of packets either in a buffer or being output, and the steady state probability of being in state i is denoted p_i. When we choose an approximate model, such as PEB, that aims to capture various effects of the real system in a single dimension, we must make certain simplifying approximations, which compromise on accuracy. The PEB approximations are mainly associated with the modeling of the packet arrival process. In the real system, when a packet arrives, it is progressively either directed to the output port, or written to the buffer, or discarded. Each of the processes takes one packet time to be completed. In any of these cases, no new packet can arrive at that particular input port until the current packet has completely arrived. In contrast, our Markov chain model assumes a state-dependent Poisson arrival process in which packets can arrive at any time, and they do so instantaneously. Without these modeling approximations we will need a large state space to capture all the relevant effects, in which case the solution will be computationally prohibitive for large problems and therefore impractical. Although we resort to approximations, we are able to capture in a single dimension Markov chain model all the important effects to a high degree of accuracy. The PEB simplifying approximations are described in detail in Table I.

The most subtle approximation is that each busy output link causes an input link to be busy simultaneously, reflected in events 4 and 5. This is the reason there is no overall change in arrival rate in the PEB at event 4. The packet that completes the service releases both an input and an output thus reducing the idle time between packet arrivals to all input ports. The new packet that comes from the buffer is modeled as simultaneously occupying an input and an output port for its entire transmission duration. This means that idle time between packet arrivals to all input ports is increased (by exactly the same amount of the previous decrease) to model the fact that the input port is unavailable. Similarly, in event 5, the PEB arrival rate increases when a packet finishes transmission, even

if the packet which departed was being sent from the buffer. This approximation cancels that in event 3, as follows.

As defined in Table I, τ is the time offset between when a packet first arrives and starts entering the buffer, and when it starts being transmitted through the output port. For a packet of transmission time m, there is a period $\min(m, \tau)$ during which the packet is being transmitted, but not causing an input port to be busy. PEB under-estimates the arrival rate during that period. However, in such a case there was an equal time in event 3 during which an input port was assumed to be idle, while in fact it was busy receiving a packet being written to the buffer. Thus, the above two effects cancel out each other from the point of view of the arrival rate (see more details in Appendix I). However, they cause a slight increase in the discard probability: the arrival rate is over-estimated in states $i \approx K + B$, when a large fraction of packets will be discarded, and the corresponding under-estimate occurs in states $i \ll K + B$ when packets will not be discarded. This causes PEB to be conservative and over-estimate the discard probability slightly.

The task is now to find state transition rates dependent on P_D. In event 2, packets that arrive at the input port are discarded. The one-dimensional model cannot keep track of the number of input ports discarding packets, and so the model allows packets to arrive at the input port immediately after a discarded packet arrives. To compensate, as in [11], the average time per arrival spent discarding packets, P_D/μ, is modelled by increasing the *effective* mean idle time, between when a packet finishes being received and when the next arrives, from $1/\lambda$ to

$$\frac{1}{\lambda^*} = \frac{1}{\underline{\lambda}^*(\lambda, P_D)} \equiv \frac{1}{\lambda} + \frac{P_D}{\mu}. \tag{5}$$

As shown in Appendix I, this yields a model whose mean arrival rate is equal to Λ, the correct rate for the "real" multidimensional Markov system.

Under the approximation that there is exactly one busy input port for each busy output port, the number of busy input ports in state i is simply $\min(i, K)$. The number of idle input ports is thus $M - \min(i, K)$, and the arrival rate in state i equals the number of idle input ports times the reciprocal of the effective mean idle time. Thus, the effective arrival rate in state i is $[M - \min(i, K)]\lambda^*$. Similarly, the departure rate in state $i+1$ is the number of busy output ports times the reciprocal of the mean busy time, namely $\min(i+1, K)\mu$. The resulting steady state equations are

$$\frac{p_{i+1}}{p_i} = \begin{cases} (M - i)\dfrac{\lambda^*}{(i+1)\mu} & 0 \le i \le K - 1 \\[2ex] (M - K)\dfrac{\lambda^*}{K\mu} & K \le i \le K + B - 1 \end{cases} \tag{6}$$

together with the normalization equation

$$\sum_{i=0}^{K+B} p_i = 1. \tag{7}$$

Recall that the mean arrival rate of the PEB model is equal to Λ, given by (3). Moreover, since packet loss only occurs when the buffer is full, the rate of packets arriving and being discarded is the arrival rate in state $K + B$, namely $[M - \min(K + B, K)]\lambda^*$, times the probability of being in that state, p_{K+B}. The packet loss probability, given by the rate of discards over the mean total arrival rate is thus

$$P_D = \frac{(M - K)\lambda^* p_{K+B}}{\Lambda}. \tag{8}$$

Notice that the steady state p_{K+B} is a function of λ^* which is actually determined by P_D. This gives rise to another fixed-point relationship. If λ is known, P_D can be obtained through an efficient algorithm provided in [11], solving the fixed-point equations of (5) and (8). Then, the closed-loop system solution can be obtained by the algorithm provided in Appendix III. The closed-loop system combines (5) and (8) with (4) into a larger set of fixed point equations. Appendix II proves the existence of a unique fixed-point solution for the closed-loop system and Appendix III proves that our algorithm converges to this fixed-point solution.

C. Related models

As well as the foregoing model, three related models will also be evaluated numerically in Section V.

The first simplified model is the *Engset with buffer* (EB), derived from PEB by assuming that the mean idle time from the point of view of the switch equals $1/\lambda$. This means that EB neglects the reduction in traffic load due to the effect of progressive packet discarding. The packet loss probability in this special case, denoted $P_D(EB)$, is obtained by substituting the approximation

$$\lambda^* \approx \lambda \quad \text{for all } 0 \le i \le K + B$$

into (6), and after using (6) and (7) to obtain π_{K+B}, $P_D(EB)$ is obtained by

$$P_D(EB) = \frac{(M - K)\lambda p_{K+B}}{\Lambda_{EB}}, \tag{9}$$

where

$$\Lambda_{EB} = \frac{M}{1/\lambda + (1 - P_D(EB))/\mu}, \tag{10}$$

which gives

$$P_D(EB) = \frac{(M - K)p_{K+B}(\mu + \lambda)}{M\mu + \lambda(M - K)p_{K+B}}. \tag{11}$$

Notice that for the case $B = 0$, the EB model reduces to the classical Engset model.

This over-estimates the arrival rate, and will yield an over-estimate of the discard probability. This effect is more apparent in high loading where discard probability is high, making higher impact on λ^* as can be seen in (5).

The second is the M/M/K/($K + B$), a "textbook" model which assumes a state-independent Poisson arrival process, and thus ignores the correlation between the arrival rate and the state of the ports. This approximation will yield accurate results only when the number of input ports is significantly larger than the number of output ports plus buffer spaces ($M \gg K + B$).

TABLE I
COMPARISON BETWEEN A PHYSICAL SYSTEM AND THE PEB MODEL

Event and corresponding switch operation	Corresponding PEB model approximation
1. A packet arrives when a free port is available. It is then progressively directed to the output port.	For the entire duration of the packet, (a) the aggregate service rate is increased to model the fact that an extra packet is being served and (b) the mean idle time between packets is increased to model the unavailability on one input link/port and the fact that no new packets can arrive at that input port.
2. A packet arrives when all output ports and buffer spaces are busy. The packet is progressively discarded from the moment it arrives.	The PEB model decouples the loss of the packet from the effect of progressive discarding process and assumes independence between the two. The packet is discarded instantaneously, so that this event causes no immediate change in the arrival rate. Instead, the long-run arrival rate of packets to all input ports is reduced by increasing the mean idle time based on the mean time to discard a packet and the long-run probability of this event to occur (i.e., packet loss probability).
3. A packet arrives when all output ports are busy but there is a vacant buffer space. The packet is then written to the buffer as it arrives.	The packet is written instantaneously to the buffer. The required compensation to the arrival rate is made at events 4 and 5 instead of here.
4. An output port becomes free, and the buffer is not empty. A packet starts being read from the buffer. **A buffer space is available to store a new packet as soon as the packet previously in that buffer space has both (a) been completely written to the buffer and (b) started to be read from the buffer.** Moreover, the packet may start being read before it has been fully received. The time offset between when the packet first arrives and when it starts being transmitted is τ. From time τ before the packet finishes being transmitted, the packet does not occupy the input port, and new packets may arrive.	A buffered packet is released instantaneously from the buffer. The transition rates of the Markov chain are unchanged, because the number of packets in service remains K.
5. An output port becomes free and the buffer is empty. The number of busy output ports decreases by one.	The aggregate service rate is decreased to model the fact that one fewer packet is being served. In addition, the mean idle time between packets is decreased to model the availability to receive packets of the port on which this packet arrived.

The numerical results for the packet loss probability of the M/M/K/($K+B$) queue in this paper were calculated using the closed form

$$P_D = \frac{K^K A^{K+B}}{K!} \left[\sum_{i=0}^{K-1} \frac{(KA)^i}{i!} + \sum_{i=K}^{K+B} \frac{(KA)^i}{K! K^{i-K}} \right]^{-1} \quad (12)$$

where, using (3),

$$A = \frac{\Lambda}{K\mu} = \frac{M\lambda}{K(\lambda + \mu)}. \quad (13)$$

Recall that λ here is the reciprocal of the mean idle time of an input port, rather than the arrival rate which is its common meaning in queueing theory. Note that, like PEB, this model has the correct overall arrival rate. However, unlike the arrival process of PEB where the arrival rate may decrease when more packets are in the switch (i increases), the arrival rate here does not decrease in this case. This causes the arrival rate in state $i = K + B$ (the highest value that i can take) to be significantly too high. Since this is the only state in which packets are discarded, this model will over-estimate the discard probability.

The third simplified model is the above-mentioned WZ model [11]. Since the WZ and PEB models differ only in the assumptions associated with buffered packets, they revert to each other for the case $B = 0$. This will be demonstrated numerically in the Section V. The assumption limiting multiple packets from one input port can cause underestimation of the total arrival rate, and hence the discard probability, unless B is negligible compared with $M - K$. In particular, if $K < M \leq K + B$ then WZ predicts $P_D = 0$. In contrast, PEB correctly predicts that $P_D > 0$.

Insight into PEB can be gained by considering its behavior in different asymptotic regimes for M, B, K, λ and P_D. One interesting regime is that in which K and M become large with $\alpha \equiv K/M$, λ and μ fixed, and

$$\hat{p} \equiv \lambda/(\lambda + \mu) < \alpha. \quad (14)$$

Similar asymptotic regimes have long been studied for other models, which do not consider the fact that discarded

packets continue to occupy an input port, or the impact of small buffers [45], [46].

Since $\hat{p} < \alpha$, the strong law of large numbers yields that the fraction of busy servers tends to a constant, and P_D tends to 0. However, we are interested in the rate of convergence to this limit. Appendix IV shows that in this regime, the asymptotic behavior of P_D is given by

$$P_D \sim \left(\frac{(1-\alpha)^{B+1}}{\alpha^B}\right)\binom{M}{K}(\hat{p})^{K+B}(1-\hat{p})^{M-K-B-1},\tag{15}$$

or more explicitly by

$$P_D \sim \sqrt{\frac{1-\alpha}{2\pi\alpha}}\left(\frac{\lambda+\mu}{\mu}\right)\left(\frac{\lambda(1-\alpha)}{\mu\alpha}\right)^B \frac{e^{-M D_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))}}{\sqrt{M}},\tag{16}$$

where $x \sim y$ denotes $x/y \to 1$, and $D_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))$ is the Kullback-Leibler divergence [47] between the binary distributions $b(\alpha)$ and $(b(\hat{p}))$ with probabilities α, $1-\alpha$ and $\hat{p}$, $1-\hat{p}$, respectively. In this case, the Kullback-Leibler divergence becomes

$$D_{\mathrm{KL}}(b(\alpha)||b(\hat{p})) = \alpha\left(\ln\frac{\alpha}{\hat{p}}\right) + (1-\alpha)\left(\ln\frac{1-\alpha}{1-\hat{p}}\right).$$

IV. EXTRAPOLATION FOR HIGH BIT RATE SIMULATIONS

It is difficult to simulate a 40 Gbit/s network, considering that many networks limit TCP packets to no larger than 1500 kBytes, and thus very many packets must be simulated. In this section, we describe a new method based on extrapolation to obtain results for packet loss probability and throughput for cases where direct simulation is computationally prohibitive. The extrapolation is based on curve fitting to obtain results for simulations that are run for cases where the capacity is sufficiently small that the simulations are computationally feasible. In all cases, the curve fitting was done using the method of least-squares [48].

A. Scaling regime for the extrapolation method

Extrapolation is most effective if the known functional dependencies are first removed. (Indeed, extrapolation of such known dependencies is often sufficient [49].) From (2),

$$\rho C_O \approx R_{agg} = N\frac{\sqrt{1.5/P_D}}{\overline{RTT}_H}\tag{17}$$

where ρ is the bottleneck utilization, and C_O is the total output bit rate.

The relationship between bottleneck utilization and packet loss probability is thus

$$\rho \approx \frac{\sqrt{1.5/P_D}}{\overline{RTT}_H\, C_O/N}.\tag{18}$$

We aim to study the trend of bottleneck utilization at different bit rates. As well as the equilibrium (18), the dynamics of a TCP connection are largely dependent on its bitrate-delay product (BDP). By reducing both C_O and N in proportion, the distribution of BDPs is maintained in each scenario. By also maintaining a constant value of $\overline{RTT}_H$, the only functional dependence that needs to be extrapolated is
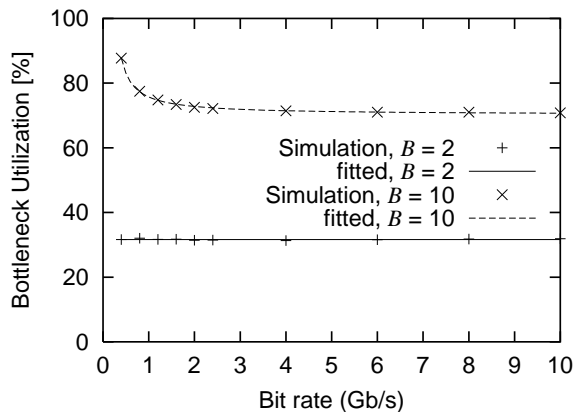

Fig. 3. Curve fitting of bottleneck utilization, $B = 2$ and 10 [packets]

the impact of increased multiplexing at higher bit rates. Since this multiplexing leads to Poisson arrivals when the number of flows is large, the throughput ρ tends to a constant in this scaling regime, allowing this extrapolation method to work up to very high bit rates.

B. Example of the extrapolation method

Consider a 40 Gbit/s link carrying 4000 TCP connections. To determine the throughput for a particular buffer size, we can extrapolate from a range of scenarios involving 0.4, 0.8, 1.2, 1.6, 2.0, 2.4, 4, 6, 8 or 10 Gbit/s. Since the share of each flow of the bottleneck bit rate is 10 Mbit/s in the 40 Gbit/s case, the same bottleneck bit rate per flow will be used for the lower bit rate cases, which will have 40, 80, 120, 160, 200, 240, 400, 600, 800 and 1000 TCP flows respectively. For each scenario, we consider long lasting flows, and we use paced TCP NewReno to reflect the fact that access links very much slower than 40 Gbit/s will space out packets. The RTTs are assumed to be uniformly distributed between 60 ms and 180 ms giving a weighted harmonic mean of $\overline{RTT}_H = (180-60)/\log_e(180/60) = 109.1$ ms. **The simulations were performed using ns2, making for simplicity the optimistic assumption that a buffer space is available as soon as the packet it contains starts transmission.**

Figure 3 shows the fitted curves for the cases where the buffer sizes are $B = 2$ and 10 packets. For $B = 2$, the points form a straight line so that the extrapolated value for the utilization at 40 Gbit/s is 31.7%. By comparison, the utilization when $B = 10$ decreases with the available bit rate. The shape of the fitted curve in this case is close to that of a hyperbola. In particular, fitting the nine simulated points gives the function $0.702 + 0.0481/(x - 0.126)$, whose standard error is 0.00673 and correlation coefficient is 0.999. The predicted utilization at 40 Gbit/s is 70.3%.

We can also predict the packet loss probability using extrapolation as illustrated in Figure 4. Instead of extrapolating the packet loss probabilities directly, we extrapolate their base-10 logarithm.

The points in Figure 4 for a buffer of 2 packets again form a straight line, giving an extrapolated value for the packet loss probability at 40 Gbit/s of $10^{-3.04}$. However, as in the case of
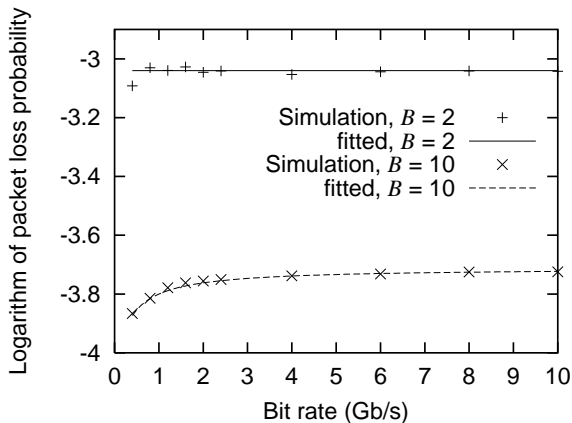
Fig. 4. Curve fitting of log(packet loss probability) for the buffer size $B = 2$, 10 [packets]

Figure 3, the shape of the curve when $B = 10$ is approximately a hyperbola and so we find the function $-3.712 - 0.117/(x + 0.327)$ that fits the points in the figure. The standard error is 0.00698 and correlation coefficient is 0.982. The extrapolated packet loss probability at 40 Gbit/s is $10^{-3.715}$.

V. MODEL VALIDATION

The open-loop and closed-loop models will now be validated. After that, they and the extrapolation technique can be used to study high bit-rate networks. In the next section, we use them to investigate the impact of DWDM on buffer requirements, and the relationship between pacing and low bit-rate access links.

A. Validation of the open-loop analytical model

In order to evaluate the accuracy of the open-loop model, its predictions will be compared with Monte Carlo simulation [50] for a wide range of parameter values. Monte Carlo simulation is used instead of ns2 because TCP is not involved here. **In our simulation here we consider the case described in Table 1 whereby a new incoming packet can enter a buffer space at any time after the previous packet has both completed being inserted into that buffer space and started being transmitted. When a channel becomes available and more than one packet is buffered, the packet sent is the one whose first bit arrived at the buffer earliest. Also, in these open-loop simulations** both the packet size and the packet interval are exponentially distributed, as in the analytical open-loop model, because the aim here is to test the approximations to the state-dependent arrival rates. Later, we test the effect of the exponential assumptions when we compare with ns2 simulations that include TCP where the packet length and interval are no longer exponentially distributed.

Results are presented here for the packet loss probability versus the normalized intended offered load [51], defined by $M\lambda/[K(\lambda + \mu)]$. In all the simulations, the runs were sufficiently long to keep the widths of the two-sided 95% confidence intervals, based on the Student-t distribution, within 3% of the average values of packet loss probability shown in the figures.

We now consider the model predictions for a wide range of loads, packet loss probabilities and numbers of ports.

Figure 5 compares the predicted packet loss probability with the simulated values for the case of $M = 2$ input ports and $K = 1$ output port. **All the models overestimate the true discard probability, with the PEB and WZ model yielding the highest accuracy.**

As mentioned in Section III.C, in Figure 5(a), the WZ reverts to the PEB model as the buffer is zero. In Figure 5(b) WZ yields $P_D = 0$ since $M < K + B$, and so it does not appear in this log-scale figure.

In both subfigures, when the traffic load is low, the packet loss probability is sufficiently small that EB is indistinguishable from PEB. However, it noticeably over-estimates the discard probability at high load, since its over-estimate of the total arrival rate becomes significant in that case as discussed in Section III.C.

The M/M/K/($K+B$) model is generally the least accurate. This is because, as explained previously, it over-estimates the arrival rate in the state $i = K+B$ in which discards can occur.

For cases with $K > 2$, such as those in Figure 6, PEB is again the most accurate, **except for the case of $B = 5$,** and close to the simulated results. At these low packet loss probabilities, EB is indistinguishable from PEB. The M/M/K/($K+B$) model, while not as accurate, is also quite close. **In fact, when $B = 5$, the M/M/K/($K+B$) model yields the highest accuracy.** The WZ model significantly under-estimates the true packet loss probability when $B = 10$, since B is not negligible compared with $M - K$.

Figure 7 shows the results for various numbers of input ports, M, for $K = 1$ output ports, a buffer of $B = 3$ packets and a normalized intended traffic load of $A = 0.4$. The PEB and EB model remains the most accurate.

Figure 8 shows results when the intended traffic load is larger than the bottleneck capacity, $A > 1$. In this case, the bottleneck is almost fully utilized, and the packet loss probability is approximately

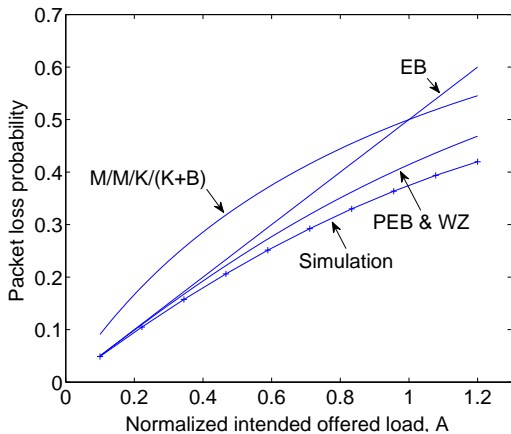$$P_D \approx \frac{A-1}{A} = \frac{M\lambda - K(\mu + \lambda)}{M\lambda}. \quad (19)$$

Models PEB, WZ and M/M/K/$K + B$ take this packet loss probability into account, and give very similar results, very close to the simulation results.

Recall that the EB model neglects the reduction in traffic load due to discarded packets assuming that packets continue to arrive at an input port while it is discarding a packet. This means it over-estimates the arrival rate λ when the packet loss probability is high. Letting $x = K\mu/[(M - K)\lambda]$, it can be shown that for $x^B \ll 1$ (large buffer or heavy overload), EB over-estimates (19) by a factor of approximately $M/(M-K)$.
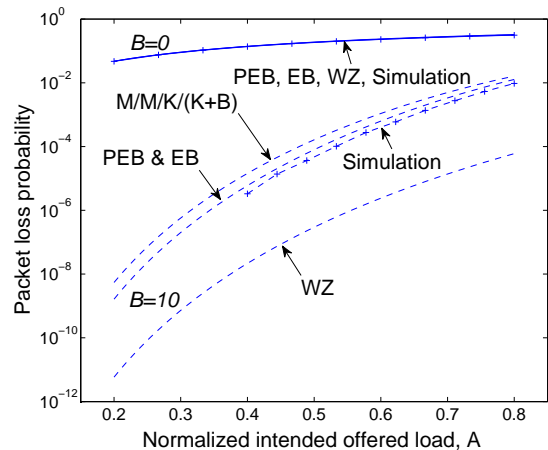
The above results, and many additional results not presented here, show that PEB is quite accurate in a wide range of scenarios. Since PEB is not computationally prohibitive and almost as computationally efficient as the other modules, it is used for the results in the following section.

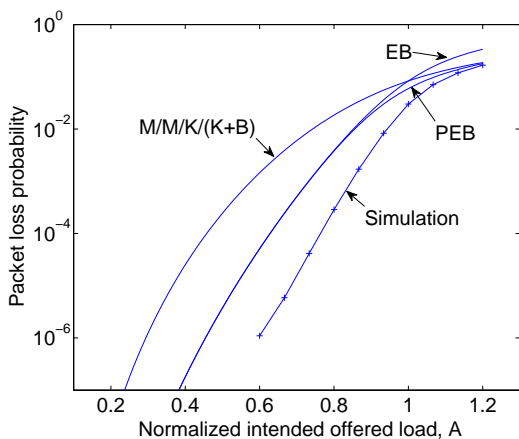### B. Validation of the closed-loop model

Figure 9 compares the results obtained based on the closed-loop PEB model with those obtained using the extrapolated
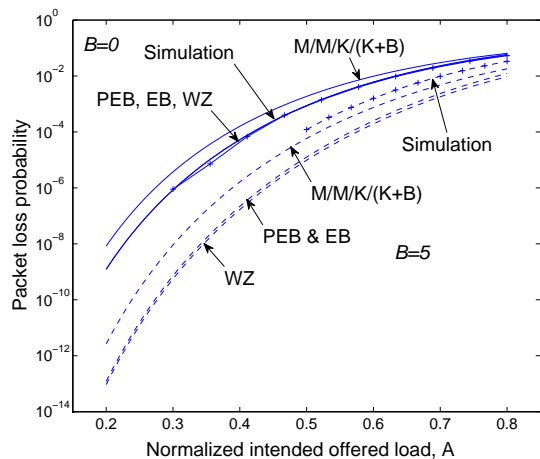
(a) $M = 2$ input ports, $K = 1$ output ports, $B = 0$ packet buffer

(b) $M = 2$ input ports, $K = 1$ output ports, $B = 10$ packet buffer

Fig. 5.   Packet loss probability vs. normalized intended traffic load

(a) $M = 16$ input ports, $K = 2$ output ports; $B = 0$ solid line, $B = 10$ dashed line

(b) $M = 80$ input ports, $K = 20$ output ports; $B = 0$ solid line, $B = 5$ dashed line

Fig. 6.   Packet loss probability vs. normalized intended traffic load.

ns2 simulations. The bit rate of every link is set at 40 Gbit/s. The number of output ports, K, varies between 1 and 100. The number of TCP flows is $4000K$. The payload of each TCP packet is 1000 Bytes. The bit rate of each access link is 100 Mbit/s.

When the buffer is very small ($B = 0$ or 1 packet), the PEB model produces very accurate and slightly conservative results, predicting slightly lower throughput than the simulations. For larger buffers ($B = 10$ packets), the PEB model becomes increasingly conservative relative to the simulation. However, the good overall agreement suggests that both approximate evaluation techniques yield good estimates.

VI. SCALING: INCREASED UTILIZATION FROM DWDM

Having developed the necessary tools, it is now possible to study ultra-high bitrate networks. Moore's law [25] states that the number of transistors on an integrated circuit doubles every 18 months, and computing power is increasing at a similar

rate. Besides CPU computing power, it has been found that the Internet traffic and switch capacity are scaling up at similar rates [26]. The following figures demonstrate the effects of scaling up various network parameters. The payload of TCP packets in all the simulations is 1000 Bytes. Unless otherwise specified, the access bit rate per connection is 100 Mbit/s.

As described in Section IV, if the bottleneck bit rate increases linearly with the number of TCP flows while other factors such as the numbers of input and output ports and the buffer size do not change, the bottleneck utilization converges to a unique limit. This also means that the buffer requirements will not change under this scaling.

Consider a DWDM system with many wavelengths. If the number of links per trunk is scaled up in proportion to the number of TCP flows while the bit rate of each link is kept constant, the bottleneck utilization will increase, due to
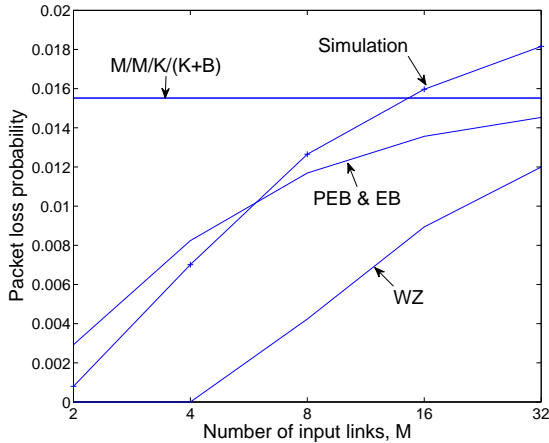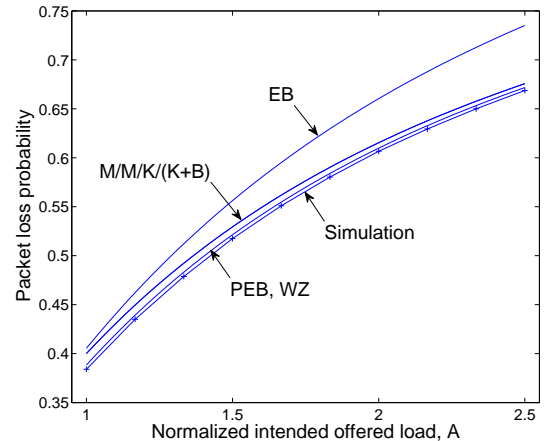
Fig. 7. Packet loss probability vs. number of input ports, for $K = 1$ output ports, $B = 3$ packet buffer, and load $A = 0.4$.

(a) $M = 16$ input ports, $K = 2$ output ports, $B = 0$ packet buffer

(b) $M = 80$ input ports, $K = 20$ output ports, $B = 5$ packet buffer

Fig. 8. Packet loss probability vs. normalized intended traffic load.

the increased statistical multiplexing reducing the packet loss probability. That is, the throughput will increase *faster* than the rate of increase of capacity. We demonstrate this effect using ns2 simulations and the extrapolation method for the following simulation scenario. The bit rate of each link is extrapolated to 40 Gbit/s. The input traffic comes from four trunks, each with the same number of links as the output trunk. No buffer is used. The number of TCP flows increases in proportion to the number of links per trunk to keep each user having 100 Mbit/s, 10 Mbit/s or 2 Mbit/s bottleneck bit rate. Notice that although the access bit rate is set at 100 Mbit/s, the bottleneck link may limit the access to a lower rate per user in which case the access link is not fully utilized. As seen in Figure 10(a), the bottleneck utilization increases with the number of links per trunk. If there are 100 output ports and 400,000 TCP flows, the utilization can be as high as 72% *with no buffer*.
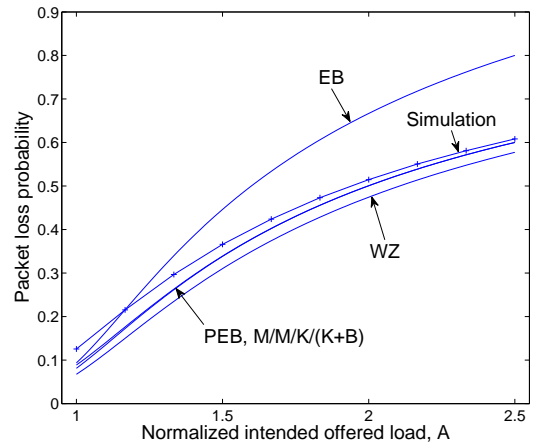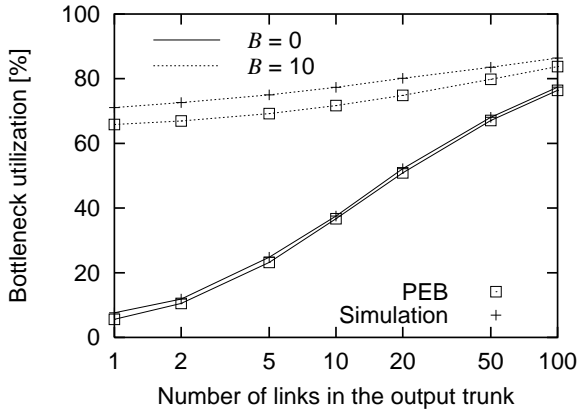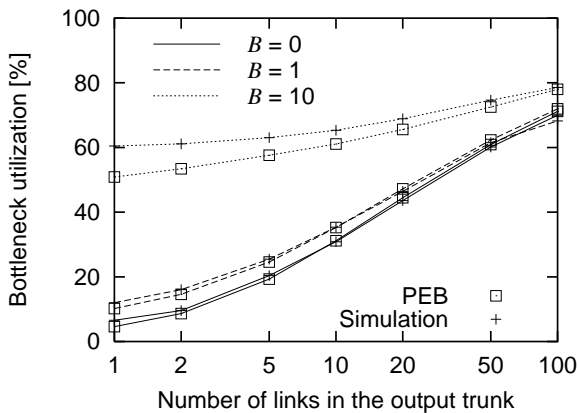
In some DWDM switches, each output port has its own buffer. Therefore, the total buffer size B in the output trunk is scaled up with the number of output ports, K. Figure 10(b) shows the link utilization in this buffered case. All the conditions in the simulations are the same as those in Figure 10(a) except that each output port possesses a one packet buffer ($B = K$). As a result, the bottleneck utilization is significantly larger than in Figure 10(a). For instance, if there are 16 output ports and 6400 TCP flows, the utilization is increased from 30% in the bufferless case to 73%. This demonstrates that even a single packet buffer per output wavelength is sufficient to give high link utilization.

Note that these results are for core routers carrying large numbers of flows. The number of flows in progress at any one time actually depends on the rates achieved by the flows [52], because achieving a low rate causes flows to last longer. We do not expect that to affect our main contention that the number of parallel DWDM channels affects the required buffer size in a core router.

Fundamentally, as explained in the following, the effect of scaling up the number of links per trunk is closely related to the comparison of having full wavelength conversion and

not having wavelength conversion. Assuming each trunk has F optical fibers, and each optical fiber can carry W wavelengths, we have FW links per trunk. In this case, as in [11], considering only traffic transmitted in a given wavelength, and assuming balanced (uniform) use of the wavelengths, the case of no wavelength conversion can be viewed as a scaled-down version of the case with full wavelength conversion, where the number of links per trunk is reduced by a factor of W. In other words, in the no-wavelength-conversion case, we consider one network with F links per trunk. Such a network is one out of W separate identical networks each of which has F links per trunk. Figure 10(a) demonstrates the deterioration of efficiency as a result of scaling down the number of links per trunk.

If the link bit rate is scaled up while the number of ports and buffer size do not vary, each TCP connection sends a higher bit rate over the bottleneck. Consider a case where there are $K = 20$ links per trunk, 4 input trunks giving $M = 4K = 80$
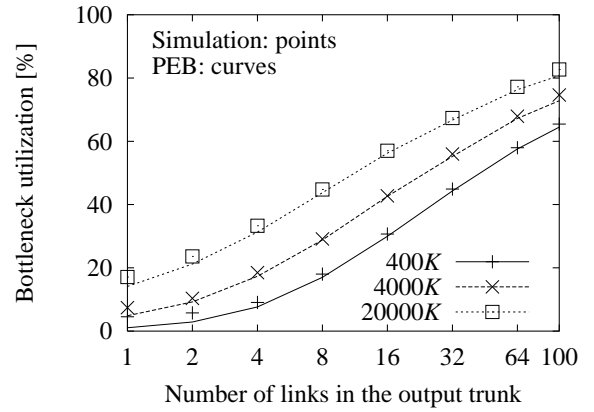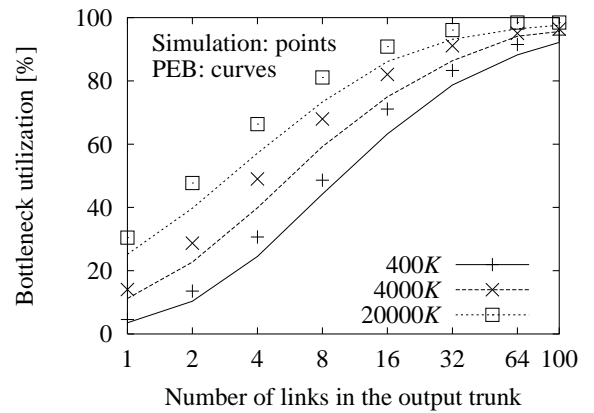
(a) $L - 1 = 2$ input trunks

(b) $L - 1 = 16$ input trunks

Fig. 9. Bottleneck utilization vs. number of links per trunk (K).

(a) $B = 0$ packet buffer

(b) One packet buffer per output port; $B = K$

Fig. 10. Bottleneck utilization at different numbers of links per trunk, for "fair share" rate per flow of 2 Mbit/s (20,000K flows), 10 Mbit/s (4,000K flows) and 100 Mbit/s (400K flows).

links, and $1000K = 20,000$ TCP flows, and the bit rate of each link varies from 1 Gbit/s to 128 Gbit/s. The throughput for such a system is shown in Figure 11. Notice that although the bit rate increases, the bottleneck utilization *decreases*. It is well accepted [1]–[3], [53] that the buffer should be increased when the bit rate increases to maintain utilization level.

A. The effects of access links and un-paced TCP

The above simulations are all based on paced TCP NewReno. However, TCP traffic in the Internet is not paced. TCP without pacing tends to send bursts made up of back-to-back packets so that the packet loss probability will be higher at the bottleneck.

Enachescu *et al.* [6] suggested that having access bit rates small compared with core link bit rate smooths the bursts so that traffic that uses regular TCP is not disadvantaged in comparison to traffic that uses paced TCP. Here we show that this effect also applies in our case of multiple links per trunk. To this end, we ran simulations for regular TCP based on our network model. The bit rate of each core link was set at 1 Gbit/s and the number of TCP flows was 400. The resulting

utilization is shown in Figure 12, in which M is the number of input ports, K is the number of output ports and B is the buffer size [packets]. It can be seen that paced TCP outperforms regular TCP only when the access bit rate is at least of the same order of magnitude as that of the core link bit rate.

Since in practice the core bit rate is typically much higher than the access bit rate, and regular TCP achieves similar throughput to paced TCP in this case, the conclusions drawn earlier in this section for paced TCP can also be applied to regular TCP.

VII. CONCLUSION

We have developed a scalable and accurate analytical method for the evaluation of TCP throughput and packet loss rate for networks containing DWDM core switches and low rate access links with large buffers. **Our method is based on a model that assumes a single network bottleneck and a fixed number of greedy sources.** Using this method, we have demonstrated the relationship between the throughput of TCP NewReno and the buffer size at a bottleneck core
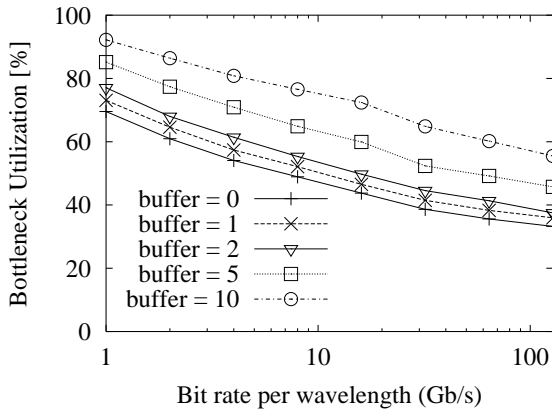
Fig. 11. Percentage bottleneck utilization as a function of link bit rate (80 input ports, 20 output ports and 20,000 TCP flows).
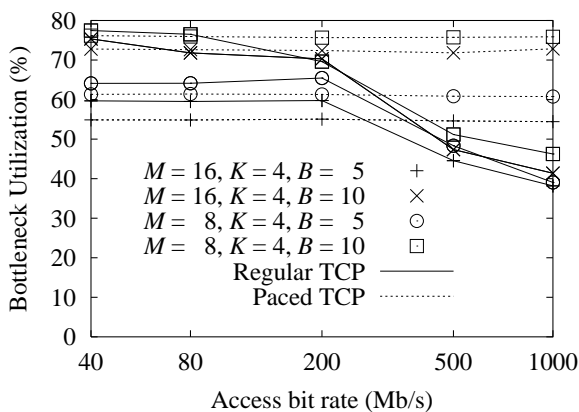

Fig. 12. Effect of access bit rate on bottleneck utilization of regular versus paced TCP over a 1 Gbit/s bottleneck.

switch considering DWDM-related parameters such as the number of links per trunk. Based on simulation results, we have demonstrated that the multiplexing provided by DWDM allows TCP to achieve utilization of up to 90% with a buffer of 10 packets, and 70% with no output buffer at all.

As part of our TCP over DWDM model, we have developed an open-loop model which on its own can provide an accurate and scalable evaluation of **DWDM packet loss rates** in cases where TCP is not used.

A thorough validation by simulations of a variety of models has been presented. We have also proposed an extrapolation method which can predict the bottleneck throughput for high bit rate switches. This is achieved by extrapolating the simulation results at smaller bit rates. Good agreement has been demonstrated between the extrapolated results and the analytical models.

We have shown that the throughput is not only determined by the buffer size but also by many other factors including the number of wavelength channels in the bottleneck trunk and the properties of the TCP traffic.

Our method can be used as part of a tool for DWDM network and switch design.

APPENDIX I: DERIVATION OF THE MEAN ARRIVAL RATE OF THE PEB MODEL

Let λ_o be the arrival rate at the switch under the PEB model. Let

$$\lambda_c = \lambda_o(1 - P_D) \tag{20}$$

be the corresponding rate at which arrivals are accepted by the switch. Now λ_c/μ is the carried load; by Little's law [40], λ_c/μ is the mean number of busy output wavelengths giving

$$\lambda_c = \mu \left(\sum_{i=0}^{K-1} i p_i + \sum_{i=K}^{K+B} K p_i \right). \tag{21}$$

The arrival rate λ_o is the probability-weighted average of the state-dependent arrival rates, namely

$$\lambda_o = \sum_{i=0}^{K-1} \lambda^*(M-i)p_i + \sum_{i=K}^{K+B} \lambda^*(M-K)p_i. \tag{22}$$

Rearranging and substituting (21) gives

$$
\begin{aligned}
\lambda_o &= \lambda^* M - \lambda^* \left(\sum_{i=0}^{K-1} i p_i + \sum_{i=K}^{K+B} K p_i \right) \\
&= \lambda^*(M - \lambda_c/\mu).
\end{aligned}
\tag{23}
$$

Substituting (20) into (23) gives

$$\lambda_o = \lambda^*[M - \lambda_o(1 - P_D)/\mu]$$

Rearranging and substituting $1/\lambda^* = 1/\lambda + P_D/\mu$ from (5) gives

$$
\begin{aligned}
\lambda_o &= \frac{M}{1/\lambda^* + (1 - P_D)/\mu} \\
&= \frac{M}{\lambda^{-1} + \mu^{-1}}
\end{aligned}
\tag{24}
$$

which is the actual arrival rate Λ, given by (3).

One interesting implication of (24) is that the effect on the arrival rate of the approximation in event 3 cancels out that of the approximation in events 4 and 5, namely that each busy output link causes an input link to be busy. This can be seen by noting that the other approximations do not change the arrival rate, as follows.

The PEB approximation of event 1 makes introduces no error to the arrival rate. The reduction in arrivals due to loss in event 2 is cancelled by the reduction in load by (5). In a period T, both add a total expected "unavailable" time $T\Lambda P_D/\mu$ to each of the M input links, reducing the number of arrivals by $T\Lambda P_D \lambda/\mu$. (The losses add idle time of $1/\mu$ for each of the $T\Lambda P_D$ loss events, while (5) adds idle time P_D/μ for each of the $T\Lambda$ arrival events.)

APPENDIX II: EXISTENCE AND UNIQUENESS OF TCP/P_D EVALUATION FIXED-POINT SOLUTION

In order to find conditions for the existence and uniqueness of the closed-loop fixed-point solution for P_D, let us first study some properties of the open-loop function from λ to P_D, denoted $P_\lambda(\lambda)$.

Making the dependence of p_i on λ^* explicit in (8), the average packet loss probability is given by

$$P_D = G(\lambda^*) \triangleq \frac{(M-K)p_{K+B}(\lambda^*)}{\sum_{i=0}^{K+B}[M-\min(i,K)]p_i(\lambda^*)}. \quad (25)$$

where $M > K > 0$. Clearly $G(\lambda^*) \in [0,1)$, since all terms are non-negative, and the denominator contains at least one positive term ($i=0$) not in the numerator.

For $1 \le i \le K+B$, (6) gives

$$
\begin{aligned}
p_i(\lambda^*) &= \frac{[M-\min(i-1,K)]\lambda^*}{\min(i,K)\mu}p_{i-1}(\lambda^*) \\
&= \left(\prod_{k=1}^{i}\alpha_k\right)(\lambda^*)^i p_0(\lambda^*),
\end{aligned}
$$

where $\alpha_i \triangleq [M-\min(i-1,K)]/[\min(i,K)\mu] > 0$ does not depend on λ^*. We write $a_i = [M-\min(i,K)]\left(\prod_{k=1}^{i}\alpha_k\right)$ and cancel $p_0(\lambda^*)$ in the numerator and denominator to obtain

$$
\begin{aligned}
G(\lambda^*) &= \frac{a_{K+B}\lambda^{*K+B}}{\sum_{i=0}^{K+B}a_i\lambda^{*i}} \\
&= \frac{1}{1+\sum_{i=0}^{K+B-1}(a_i/a_{K+B})(\lambda^*)^{i-K-B}}. \quad (26)
\end{aligned}
$$

Since $a_i/a_{K+B} > 0$ for all i, the denominator is strictly positive. It is clear that the denominator is decreasing in λ^* and so $G(\lambda^*)$ is increasing.

We observe that equations (5) to (8) provide an implicit functional form for P_D in terms of λ. Substituting $P_D = G(\lambda^*)$ from (25) into

$$\frac{1}{\lambda^*} = \frac{1}{\lambda} + \frac{P_D}{\mu}, \quad (27)$$

we can define $H:(0,\infty)\to\mathbb{R}$ such that

$$\frac{1}{\lambda} = H(\lambda^*) \triangleq \frac{1}{\lambda^*} - \frac{G(\lambda^*)}{\mu}. \quad (28)$$

Each of the terms on the right side of this equation is well defined, continuous, and strictly decreasing on the domain $\lambda^* > 0$, so that the same is true of the function $H(\cdot)$. Moreover, $H(\lambda^*)$ approaches $+\infty$ as $\lambda^* \to 0^+$, and approaches $-1/\mu$ as $\lambda^* \to +\infty$. Thus the inverse $H^{-1}:(-1/\mu,\infty)\to(0,\infty)$ is well defined, and

$$\lambda^* = H^{-1}\left(\frac{1}{\lambda}\right). \quad (29)$$

Note that H^{-1} is increasing and continuous for $\lambda > 0$. To see that it is continuous, observe that $H(\lambda^*)$ is, in fact, continuously differentiable for $\lambda^* > 0$, since $G(\lambda^*)$ is. Moreover, the derivative of $H(\lambda^*)$ is non-vanishing for $\lambda^* > 0$. The one-dimensional case of the inverse function theorem now yields that H^{-1} is also continuously differentiable and, a fortiori, continuous.

Define $P_\lambda:[0,\infty)\to\mathbb{R}$ such that

$$P_D = P_\lambda(\lambda) = G(H^{-1}(1/\lambda)). \quad (30)$$

This can be evaluated numerically by solving the "open-loop" fixed point equation. This is defined for all $\lambda > 0$, continuous and increasing, and takes values between $P_\lambda(0) = 0$ and

$$P_\lambda(\infty) = G(H^{-1}(0)). \quad (31)$$

Theorem 1: Let $\tilde{P} = 1.5(N/(M\mu RTT))^2$. If

$$P_\lambda(\infty) > \tilde{P} \quad (32)$$

then there exists a unique solution $P_D^\dagger$ to the closed-loop fixed point equation $P_D = P_\lambda(T(P_D))$ with $T(P_D) \ge 0$, and this solution satisfies

$$\tilde{P} < P_D^\dagger < P_\lambda(\infty) \quad (33)$$

Otherwise, there is no solution with $T(P_D) \ge 0$.

Proof: Any solution must with $T(P_D) > 0$ must satisfy (33), since $T(P_D^\dagger) > 0$ only if $P_D^\dagger > \tilde{P}$, while for all λ, $P_\lambda(\lambda) < P_\lambda(\infty)$. This establishes the necessity of (32).

Conversely, if (32) holds then there is a $\tilde{\lambda}$ such that $P_\lambda(\tilde{\lambda}) = \tilde{P}$. Analogously to (4), define $f:(0,\infty)\to\mathbb{R}$ by

$$f(\lambda) = \frac{N\mu\sqrt{1.5}}{M\mu RTT\sqrt{P_\lambda(\lambda)} - N\sqrt{1.5}} - \lambda. \quad (34)$$

It remains to prove that the existence of $\tilde{\lambda}$ implies there is a unique solution to $f(\lambda) = 0$ with $\lambda > 0$. Note that $f(\lambda)$ is continuous for $\lambda > 0$ except at $\tilde{\lambda}$ where the denominator vanishes. Clearly, $f(\lambda) < 0$ for $\lambda \in [0,\tilde{\lambda})$, and so there is no solution to $f(\lambda) = 0$ in that range. For $\lambda > \tilde{\lambda}$, $f(\lambda)$ is well-defined, continuous and strictly decreasing because of the properties of $P_\lambda(\lambda)$. Moreover, $f(\lambda) \to +\infty$ as λ decreases to $\tilde{\lambda}$. On the other hand, when λ becomes very large, $f(\lambda)$ becomes negative. To see this, note that the first part of it is always positive and finite for $\lambda > \tilde{\lambda}$ and decreasing in λ. Thus the second term $(-\lambda)$ dominates. The intermediate value theorem now guarantees a solution. The fact that f is strictly decreasing in the interval $(\tilde{\lambda}, +\infty)$ now yields uniqueness. ∎

It is possible to consider other discard models, P_λ. The above proof demonstrates that there will always exist a unique solution to the resulting fixed-point equations provided that the model satisfies (32) and P_λ is continuous and increasing. As well as PEB, this is also the case for EB, WZ and M/M/K/($K+B$) in this paper.

APPENDIX III: SOLVING THE FIXED-POINT BY BINARY SEARCH AND PROVING ITS CONVERGENCE

The closed-loop fixed point equation for discard probability is $P_D = \Gamma(P_D)$, where $\Gamma(p) \triangleq P_\lambda(T(p))$ for $\tilde{P} < p < P_\lambda(\infty)$. Note that evaluating $\Gamma(p)$ involves a numerical solution of (30), which will not be exact. Algorithm 1 finds the unique solution of $p = \Gamma(p)$ to within ϵ, provided that each iteration finds an approximation $\hat{\Gamma}(p)$ within $\epsilon/2$ of $\Gamma(p)$.

Since $T'(p) < 0$ for $p > \tilde{P}$ and $P_\lambda'(\lambda) > 0$ for $\lambda > 0$, $\Gamma'(p) < 0$ for $\tilde{P} < p < P_\lambda(\infty)$.

By induction,

$$P_D^\dagger \in [p^- - \epsilon/2, p^+ + \epsilon/2] \quad (35)$$

after each iteration. The base case follows from Theorem 1. Consider without loss of generality the case the $p < P_D^\dagger$

Algorithm 1 Calculate solution of $p = \Gamma(p)$ to within ϵ

1: $p^- \leftarrow \tilde{P}$, $p^+ \leftarrow P_\lambda(\infty)$	Initial bounds
2: **while** $p^+ - p^- > \epsilon$ **do**	
3: $\quad p \leftarrow (p^+ + p^-)/2$	Halve the search interval
4: $\quad$ **if** $\hat{\Gamma}(p) > p$ **then**	
5: $\qquad p^- \leftarrow p$	Tighten lower bound
6: $\quad$ **else**	
7: $\qquad p^+ \leftarrow p$	Tighten upper bound
8: $\quad$ **end if**	
9: **end while**	
10: **return** $(p^+ + p^-)/2$	ϵ satisfied, thus return x

at step 4. Since Γ is decreasing, $p < P_D^\dagger < \Gamma(p)$, whence $\hat{\Gamma}(p) > P_D^\dagger - \epsilon/2$. Two cases must be considered. If $\hat{\Gamma}(p) > p$ then the algorithm correctly updates p^-, and (35) again holds. Otherwise,

$$p > \hat{\Gamma}(p) > P_D^\dagger - \epsilon/2 \tag{36}$$

and the algorithm mistakenly updates p^+. However, adding $\epsilon/2$ to each side of (36) shows that the new p^+ still satisfies (35).

The algorithm terminates after finitely many steps when $p^+ - p^- \leq \epsilon$. The return value $(p^+ + p^-)/2$ is at most ϵ above the lower bound $p^- - \epsilon/2$ on $P_D^\dagger$, and at most ϵ below the upper bound $p^+ + \epsilon/2$.

This argument also applies to errors of up to $\epsilon/2$ in the initial value of p^+ in step 1. However, if the initial p^- is below $\tilde{P}$ then the monotonicity of Γ is no longer guaranteed and the proof does not apply.

This algorithm can be used by any other open-loop model P_λ which is continuous and increasing, provided that (32) is satisfied. Again, this is also the case for EB, WZ and M/M/K/($K + B$) in this paper.

APPENDIX IV: DERIVATION OF (15) AND (16)

First, consider the explicit expression for the occupancy probabilities, p_i, under PEB. By (6) and (7),

$$p_i = p_0 \binom{M}{i} \left(\frac{\lambda^*}{\mu}\right)^i \qquad \text{for } i = 0, 1, \dots, K-1 \tag{37a}$$

$$p_{K+i} = p_0 \binom{M}{K} \left(\frac{\lambda^*}{\mu}\right)^K (V\lambda^*)^i \qquad \text{for } i = 0, 1, \dots, B \tag{37b}$$

where

$$V = \frac{(M-K)}{K\mu} \tag{37c}$$

$$1/p_0 = S_1(\lambda^*) + S_2(\lambda^*) \tag{37d}$$

$$S_1(\lambda^*) = \sum_{i=0}^{K-1} \binom{M}{i} \left(\frac{\lambda^*}{\mu}\right)^i \tag{37e}$$

$$S_2(\lambda^*) = \sum_{i=0}^{B} \binom{M}{K} \left(\frac{\lambda^*}{\mu}\right)^K (V\lambda^*)^i$$

$$= \binom{M}{K} \left(\frac{\lambda^*}{\mu}\right)^K \frac{V\lambda^* - (V\lambda^*)^{B+1}}{1 - V\lambda^*}. \tag{37f}$$

Let $p_{K+B}(\cdot)$ be the function from λ^* to p_{K+B}, given by

$$p_{K+B}(\lambda^*) = \frac{\binom{M}{K}(\lambda^*/\mu)^K (V\lambda^*)^B}{S_1(\lambda^*) + S_2(\lambda^*)}. \tag{38}$$

Note that, since $p_{K+B}(\cdot)$ is a positive multiple of $(\cdot)^{K+B}$ divided by a positive-coefficient polynomial of degree $K + B$, it is increasing on $(0, \infty)$. As $\lambda > \lambda^*$ by (5), $p_{K+B}(\lambda) > p_{K+B}(\lambda^*)$.

Evaluating of (38) at λ and dividing numerator and denominator by $(1 + \lambda/\mu)^M$ gives, after some algebra,

$$p_{K+B}(\lambda) = \frac{\binom{M}{K}\hat{p}^K(1-\hat{p})^{M-K}(V\lambda)^B}{D_1(\lambda) + D_2(\lambda)} \tag{39}$$

where $\hat{p}$ is defined in (14),

$$D_1(\lambda) = \sum_{i=0}^{K} \binom{M}{i}(\hat{p})^i(1-\hat{p})^{M-i}$$

and

$$D_2(\lambda) = \binom{M}{K}(\hat{p})^K(1-\hat{p})^{M-K}\frac{V\lambda - (V\lambda)^{B+1}}{1 - V\lambda}.$$

Lemma 1: Consider the limit as $K \to \infty$ and $M \to \infty$ with $\alpha = K/M$, λ and μ fixed, and $\hat{p} < \alpha$. Then

$$p_{K+B}(\lambda^*) \sim \binom{M}{K}(\hat{p})^K(1-\hat{p})^{M-K}(V\lambda)^B. \tag{40}$$

$$\sim \frac{\exp(-M D_{\text{KL}}(b(\alpha)\|b(\hat{p})))}{\sqrt{2\pi M\alpha(1-\alpha)}}.$$

Proof: Notice that $D_1(\lambda)$ is the probability that a binomial random variable, say X, with mean $\hat{p}M$ and standard deviation $\sqrt{\hat{p}(1-\hat{p})M}$ satisfies $P[X \leq K]$, where $K = \alpha M$ for some fixed α, and $\hat{p} < \alpha$. By Tchebycheff's inequality

$$P[X \leq K] > P[|X - \hat{p}M| \leq K - M\hat{p}] \tag{41}$$

$$> 1 - \frac{M\hat{p}(1-\hat{p})}{(K - M\hat{p})^2} = 1 - \frac{1}{M} \cdot \frac{\hat{p}(1-\hat{p})}{(\alpha - \hat{p})^2}.$$

Similarly, $D_2(\lambda)$ and the numerator of $p_{K+B}(\lambda)$ are just constant (in M) multiples of $P[X = K] \leq P[X \geq K]$ and so tend to 0 as $O(1/M)$ by Tchebycheff's inequality. It follows that

$$p_{K+B}(\lambda) = O\left(\frac{1}{M}\right), \tag{42}$$

and since $p_{K+B}(\lambda) > p_{K+B}(\lambda^*) \geq 0$, it follows that $p_{K+B}(\lambda^*) = O(1/M)$.

To obtain more precise estimates of the probabilities, we will need some estimates of the binomial terms that we have been unable to find in the literature. Specifically, we are interested in

$$\binom{M}{K}\hat{p}^K(1-\hat{p})^{M-K} \tag{43}$$

where $K = \alpha M$ and $\hat{p} < \alpha$. We compare (43) with $\binom{M}{K}\alpha^K(1-\alpha)^{M-K}$, which, being the central term in the binomial distribution, is asymptotically $1/\sqrt{2\pi M\alpha(1-\alpha)}$ by the central limit theorem. Their quotient is

$$\frac{\alpha^K(1-\alpha)^{M-K}}{\hat{p}^K(1-\hat{p})^{M-K}} \tag{44}$$

We rewrite this as

$$\left(\frac{\alpha^\alpha(1-\alpha)^{1-\alpha}}{\hat{p}^\alpha(1-\hat{p})^{1-\alpha}}\right)^M \tag{45}$$

Evidently the behaviour of the quotient as $M \to \infty$ depends on the value of the quantity in brackets. We take the logarithm and rearrange to obtain

$$\alpha\big[\ln\alpha - \ln\hat{p}\big] + (1-\alpha)\big[\ln(1-\alpha) - \ln(1-\hat{p})\big]. \tag{46}$$

This is the Kullback-Leibler divergence $D_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))$ between the two distributions $b(\alpha)$ and $b(\hat{p})$, where $b(q)$ is the binary distribution which takes two values with probability q and $1-q$. It is well-known [47] that the Kullback-Leibler divergence is always non-negative and zero only if the two distributions are the same, though in this case the result can be achieved by a simple calculus argument. Thus $D_{\mathrm{KL}}(b(\alpha)||b(\hat{p})) > 0$ for $\hat{p} < \alpha$. This gives that (44) is asymptotically $e^{MD_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))}$, and so that

$$\binom{M}{K}\hat{p}^K(1-\hat{p})^{M-K} \sim \frac{1}{\sqrt{2\pi M\alpha(1-\alpha)}}e^{-MD_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))}. \tag{47}$$

As a result

$$p_{K+B}(\lambda) \sim (V\lambda)^B \frac{1}{\sqrt{2\pi M\alpha(1-\alpha)}}e^{-MD_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))}. \tag{48}$$

By (8) and (3), we have

$$P_D = \frac{(M-K)(\lambda+\mu)\lambda^*}{M\lambda\mu}p_{K+B}(\lambda^*), \tag{49}$$

so that

$$P_D < C_1(1-\alpha)\frac{(\lambda+\mu)\lambda^*}{\lambda\mu}(V\lambda)^B\left(\frac{e^{-MD_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))}}{\sqrt{2\pi M\alpha(1-\alpha)}}\right), \tag{50}$$

for some positive constant C_1 Now recall that

$$\frac{1}{\lambda^*} = \frac{1}{\lambda} + \frac{P_D}{\mu}, \tag{51}$$

so that if we write $\hat{p}^* = \lambda^*/(\lambda^*+\mu)$,

$$\frac{1}{\hat{p}^*} - \frac{1}{\hat{p}} = \frac{\mu}{\lambda^*} - \frac{\mu}{\lambda} \tag{52}$$

$$= P_D < C_2\frac{1}{\sqrt{M}}e^{-MD_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))},$$

for some positive constant C_2, where the inequality follows from (50) and the fact that λ^* is bounded above by the constant λ.

Now to complete the proof we need to compare $p_{K+B}(\lambda)$ and $p_{K+B}(\lambda^*)$. Although $\lambda^*/\lambda \to 1$ as $M \to \infty$, it is not immediately obvious that $p_{K+B}(\lambda^*)/p_{K+B}(\lambda)$ also tends to 1. To see that it does, first note that in the light of (52), if $K/M > \hat{p}$ then $K/M > \hat{p}^*$ for large enough M, so that another Tchebycheff calculation yields that $D_1(\lambda^*) + D_2(\lambda^*) \to 1$ as $M \to \infty$. It follows that the quotient is asymptotically,

$$\left(\frac{\hat{p}^*}{\hat{p}}\right)^K\left(\frac{1-\hat{p}^*}{1-\hat{p}}\right)^{M-K}\left(\frac{\lambda^*}{\lambda}\right)^B. \tag{53}$$

Since

$$\frac{\lambda^*}{\lambda} = 1 - \frac{\lambda^*P_D}{\mu}, \tag{54}$$

the final factor in (53) tends to 1 (exponentially fast) and can be ignored. The remainder of (53) can be rewritten as

$$\left(\frac{1/\hat{p}}{1/\hat{p}^*}\right)^M\left(\frac{(1/\hat{p}^*)-1}{(1/\hat{p})-1}\right)^{M-K}. \tag{55}$$

By equation (52), this can be expressed as

$$(1-\hat{p}^*P_D)^M\left(1+\frac{P_D}{(1/\hat{p})-1}\right)^{M-K}. \tag{56}$$

Taking log of each factor, using $\log(1+x) \leq x$ and substituting (52) yields

$$M\log(1-\hat{p}^*P_D) \geq -C\sqrt{M}e^{-MD_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))} \tag{57}$$

and

$$M(1-\alpha)\log\left(1+\frac{P_D}{(1/\hat{p})-1}\right) \leq C'\sqrt{M}e^{-MD_{\mathrm{KL}}(b(\alpha)||b(\hat{p}))} \tag{58}$$

where again C and C' are positive constants in M. Each of the right hand sides tends to 0 as $M \to \infty$. This yields that the quotient tends to 1 and, together with (47), completes the proof of (40). ∎

Combining Lemma 1 and (8) with the fact that $\lambda^*/\lambda \to 1$ as $M \to \infty$ gives (15) and (16).

- The careful analysis above makes it relatively easy to obtain second order asymptotics if necessary.

REFERENCES

[1] V. Jacobson, "[e2e] re: Latest TCP measurements thoughts." Mar. 1988, posting to the end-to-end mailing list.

[2] C. Villamizar and C. Song, "High performance TCP in ANSNET," *ACM Computer Communications Review*, vol. 24, no. 5, pp. 45–60, Oct. 1994.

[3] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," in *Proc. ACM SIGCOMM'04*, vol. 1, Portland, OR, August–September 2004, pp. 281–292.

[4] D. Wischik and N. McKeown, "Part I: Buffer sizes for core routers," *ACM Computer Communications Review*, vol. 35, no. 3, pp. 75–78, July 2005.

[5] G. Raina, D. Towsley, and D. Wischik, "Part II: Control theory for buffer sizing," *ACM Computer Communication Review*, vol. 35, no. 3, pp. 79–82, 2005.

[6] M. Enachescu, Y. Ganjali, A. Goel, and N. McKeown, "Part III: Routers with very small buffer," *ACM Computer Communication Review*, vol. 35, no. 3, pp. 83–90, July 2005.

[7] N. Beheshti, Y. Ganjali, R. Rajaduray, and D. Blumenthal, "Buffer sizing in all-optical packet switches," in *Proc. IEEE/OSA OFC/NFOEC'06*, Anaheim, CA, Mar. 2006, p. 3.

[8] C. A. Brackett, "Dense wavelength division multiplexing networks: principles and applications," *IEEE J. Select. Areas Commun.*, vol. 8, no. 6, pp. 948–964, Aug. 1990.

[9] V. Eramo and M. Listanti, "Packet loss in a bufferless optical WDM switch employing shared tuneable wavelength converters," *J. Lightwave Technol.*, vol. 18, no. 12, pp. 1818–1833, Dec. 2000.

[10] V. Eramo, M. Listanti, and A. Germoni, "Cost evaluation of optical packet switches equipped with limited-range and full-range converters for contention resolution," *J. Lightwave Technol.*, vol. 28, no. 4, pp. 390–407, Feb. 2008.

[11] E. W. M. Wong and M. Zukerman, "Bandwidth and buffer tradeoffs in optical packet switching," *J. Lightwave Technol.*, vol. 24, no. 12, pp. 4790–4798, Dec. 2006.

[12] X. Zhu and J. M. Khan, "Queuing models of optical delay lines in synchronous and asynchronous optical packet-switching networks," *Optical Engineering*, vol. 42, pp. 1741–1748, 2003.

[13] G. Das, R. S. Tucker, C. Leckie, and K. Hinton, "Multiple-input single-output FIFO optical buffers with controllable fractional delay lines," submitted for publication, [Online]. Available: http://www.ee.unimelb.edu.au/people/gdas/paper1.pdf.

[14] J. Kim, J. Choi, M. Kang, and J.-K. K. Rhee, "Design of novel passive optical switching system using shared wavelength conversion with electrical buffer," *IEICE Electronics Express*, vol. 3, no. 24, pp. 546–551, Dec. 2006.

[15] R. S. Tucker, "The role of optics and electronics in high-capacity routers," *J. Lightwave Technol.*, vol. 24, no. 12, pp. 4655–4673, Dec. 2006.

[16] S.-T. Chuang, A. Goel, N. McKeown, and B. Prabhakar, "Mathcing output queueing with a combined input output queued switch," *IEEE J. Select Areas Commun.*, vol. 17, no. 6, pp. 1030–1039, June 1999.

[17] Z. Rosberg, H. L. Vu, M. Zukerman, and J. White, "Performance analyses of optical burst switching networks," *IEEE J. Select. Areas Commun.*, vol. 21, no. 7, pp. 1187–1197, Sept. 2003.

[18] Z. Rosberg, A. Zalesky, H. L. Vu, and M. Zukerman, "Analysis of OBS networks with limited wavelength conversion," *IEEE/ACM Trans. Networking*, vol. 14, no. 5, pp. 1118–1127, Oct. 2006.

[19] A. Zalesky, H. L. Vu, Z. Rosberg, E. W. M. Wong, and M. Zukerman, "Stabilizing deflection routing in optical burst switched networks," *IEEE J. Select. Areas Commun., Suppl. Opt. Commun. Netw.*, vol. 25, no. 6, pp. 3–19, Aug. 2007.

[20] R. Cavanaugh, H. Newman, F. van Lingen, I. Legrand, Y. Xia, D. Nae, C. Steenberg, S. Ravot, M. Thomas, J. Bunn, *et al.*, "UltraLight: An ultrascale information system for data intensive research," in *Proc. CHEP'06*, Mumbai, India, Feb. 2006.

[21] Cisco Systems, Inc., "Approaching the zettabyte era," June 16 2008. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481374_ns827_Networking_Solutions_White_Paper.html

[22] M. Mathis, J. Semke, J. Madhavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," *ACM Computer Cmmunication Review*, vol. 27, no. 3, pp. 67–82, July 1997.

[23] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling TCP Reno performance: A simple model and its empirical validation," *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp. 133–145, Apr. 2000.

[24] Information Sciences Institute, University of South California, "The *ns* simulator and the documentation," [on-line]. Available: http://www.isi.edu/nsnam/ns/.

[25] R. R. Schaller, "Moore's law: past, present and future," *IEEE Spectr.*, vol. 34, no. 6, pp. 52–59, June 1997.

[26] J. Nielsen, "Nielsen's law of Internet bandwidth," Apr. 1998, [on-line]. Available: http://www.useit.com/alertbox/980405.html.

[27] C. Brezinski and M. Redivo, *Extrapolation Methods. Theory and Practice.* Zaglia, North-Holland, 1991.

[28] J. Kulik, R. Coulter, D. Rockwell, and C. Partridge, "A simulation study of paced TCP," BBN Technologies, BBN Technical Memorandum 1218, Aug. 1999.

[29] Y.-S. Choi, K.-W. Lee, T.-M. Han, and Y.-Z. Cho, "High-speed TCP protocols with pacing for fairness and TCP friendliness," in *Proc. IEEE TENCON'04*, vol. C, Chiang Mai, Thailand, Nov. 2004, pp. 13–16.

[30] M. Allman, V. Paxson, and W. Stevens, "RFC 2581 - TCP congestion control," Apr. 1999, [on-line]. Available: http://www.faqs.org/rfcs/rfc2581.html.

[31] S. Floyd and T. Henderson, "RFC 2582 - The NewReno modification to TCP's fast recovery algorithm," Apr. 1999, [on-line]. Available: http://www.faqs.org/rfcs/rfc2582.html.

[32] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "RFC 2018 - TCP selective acknowledgment options," Oct. 1996, [on-line]. Available: http://www.faqs.org/rfcs/rfc2018.html.

[33] Y. Gu, D. Towsley, C. V. Hollot, and H. Zhang, "Conestion control for small buffer high speed networks," in *Proc. IEEE INFOCOM*, Anchorage, AK, May 2007, pp. 1037–1045.

[34] J. Hui, *Switching and Traffic Theory for Integrated Broadband Networks.* Kluwer Academic Press, 1990.

[35] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, vol. 2, no. 1, pp. 1–15, February 1994.

[36] V. Paxson and S. Floyd, "Wide-area traffic: The failure of Poisson modeling," *IEEE/ACM Trans. Networking*, vol. 3, no. 3, pp. 226–244, June 1995.

[37] R. G. Addie, M. Zukerman, and T. D. Neame, "Fractal traffic: Measurements, modelling and performance evaluation," in *Proc. IEEE INFOCOM '95*, vol. 3, April 1995, pp. 977–984.

[38] ——, "Broadband traffic modeling: simple solutions to hard problems," *IEEE Commun. Mag.*, vol. 36, no. 8, pp. 88–95, August 1998.

[39] T. Engset, "Die Wahrscheinlichkeitsrechnung zur Bestimmung der Wähleranzahl in automatischen Fernsprechämtern," *Elektrotechnische Zeitschrift*, vol. 39, no. 31, pp. 304–306, Aug. 1918.

[40] M. Zukerman, "An introduction to queueing theory and stochastic teletraffic models," May 2007, [on-line]. Available: http://www.ee.unimelb.edu.au/staff/mzu/classnotes.pdf.

[41] J. W. Cohen, "The generalized Engset formulae," *Philips Telecommunication Review*, vol. 18, no. 4, pp. 158–170, Nov. 1957.

[42] E. W. M. Wong, A. Zalesky, and M. Zukerman, "On generalizations of the Engset model," *IEEE Commun. Lett.*, vol. 11, no. 4, pp. 360–362, Apr. 2007.

[43] R. Syski, *Introduction to Congestion Theory in Telephone Systems*, 2nd ed. North Holland, 1986.

[44] N. Akar and Y. Gunalay, "Stochastic analysis of finite population bufferless multiplexing in optical packet/burst switching systems," *IEICE Transactions on Communications*, vol. E90-B, no. 2, pp. 342–345, Feb. 2007.

[45] D. Mitra and J. Morrison, "Erlang capacity and uniform approximation of shared unbuffered resources," *IEEE/ACM Trans. Networking*, vol. 2, no. 6, pp. 558–570, December 1994.

[46] A. W. Berger and W. Whitt, "Effective bandwidths with priorities," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 447–460, August 1998.

[47] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 79–86, 1951.

[48] S. Chapra and R. Canale, *Numerical methods for engineers.* McGraw-Hill, New York, 1988.

[49] R. Pan, B. Prabhakar, K. Psounis, and D. Wischik, "SHRINK: A method for scalable performance prediction and efficient network simulation," in *Proc. IEEE INFOCOM*, 2003.

[50] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, 2nd ed. Springer-Verlag, New York, 2004.

[51] H. Akimaru and K. Kawashima, *Teletraffic – Theory and application.* Springer, London, 1999.

[52] A. Lakshmikantha and R. Srikant, "Impact of file arrivals and departures on buffer sizing in core routers," in *Proc. IEEE INFOCOM*, Apr. 2008.

[53] L. L. H. Andrew, T. Cui, J. Sun, M. Zukerman, K.-T. Ko, and S. Chan, "Buffer sizing for nonhomogeneous TCP sources," *IEEE Commun. Lett.*, vol. 9, no. 6, pp. 567–569, June 2005.