Towards a Bufferless Optical Internet

<table>
<thead>
<tr>
<th>Journal:</th>
<th><em>Journal of Lightwave Technology</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>Manuscript ID:</td>
<td>JLT-10359-2008.R2</td>
</tr>
<tr>
<td>Manuscript Type:</td>
<td>Original Paper</td>
</tr>
<tr>
<td>Date Submitted by the Author:</td>
<td>n/a</td>
</tr>
<tr>
<td>Complete List of Authors:</td>
<td>Wong, Eric; City University of Hong Kong, Electronic Engineering Andrew, Lachlan; California Institute of Technology, Department of Computer Science Cui, Tony; The University of Melbourne, Electrical and Electronic Engineering Department Moran, Bill; The University of Melbourne, Electrical and Electronic Engineering Department Zalesky, Andrew; The University of Melbourne, National Neuroscience Facility Tucker (PTL AE), Rodney; University of Melbourne, Dept. of EEE Zukerman, Moshe; The University of Melbourne, Electrical and Electronic Engineering Department</td>
</tr>
<tr>
<td>Key Words:</td>
<td>Optical switches, Optical fiber communication, Approximation methods</td>
</tr>
</tbody>
</table>
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 a clear and accurate approximation 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, $B = C_0 \times R_T T$, in order to fully utilize bottleneck links. Here, $B$ is the size of the buffer, $C_0$ is the capacity of the bottleneck link, and $R_T T$ 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 $B = C_0 \times \frac{R_T T}{\sqrt{N}}$ 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. Enachescu 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, Beheshi 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

This work was supported by the Australian Research Council and a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 121507). This material is based upon work supported by the National Science Foundation under Grant No. EIA-0303620 (WAN-in-Lab project).

E. Wong and M. Zukerman are with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR, China (e-mail: ewong@ee.cityu.hk, m.zu@cityu.hk).

L. Andrew is with the Centre for Advanced Internet Architectures (CAIA), Swinburne University of Technology, Melbourne, Australia (e-mail: Landrew AT iee.org). T. Cui, B. Moran and R. Tucker are with the Australian Research Council Special Research Center for Ultra-Broadband Information Networks (CUBIN), Electrical and Electronic Engineering Department, The University of Melbourne, Victoria 3010, Australia (e-mail: t.cui.b.moran.r.tucker@ee.unimelb.edu.au).

A. Zalesky is an ARC Postdoctoral Fellow at the University of Melbourne (LX0882154 & DP0986320); e-mail: azalesky@unimelb.edu.au. This work was conducted when M. Zukerman was with CUBIN.
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.
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 analyzing two interconnected models: one is a novel “open-loop” queuing 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 queuing 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

$$r = \frac{\sqrt{1.5}/P_D}{RTT} \approx \frac{1.22}{\sqrt{RTT}}$$

(1)

This equation follows from a family of TCP protocols, called by us “Reno”, including Reno [30], NewReno [31] and SACK [32], which are all the 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 queuing 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

$$R_{agg} = \frac{N\sqrt{1.5}/\sqrt{P_D}}{RTT_H}$$

(2)

where $R_{agg}$ [packets/sec.] is the aggregate rate of TCP flows, $N$ is the total number of TCP flows and $RTT_H$ is the weighted harmonic mean round trip time of these $N$ TCP flows, namely $RTT_H = \frac{1}{N} \sum_i \frac{r_i}{RTT_i}$, where $r_i$ and $RTT_i$ are the rate and round trip time of the $i$th flow, and the sum is taken over all flows.

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

$$\Lambda = \frac{M}{\lambda^{-1} + \mu^{-1}},$$

(3)

where the $\mu^{-1}$ reflects the reduction in load caused by each arrival, whether the packet is buffered, discarded or sent immediately, and $\Lambda$ 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 $\lambda$ and $P_D$ is $\lambda = T(P_D)$ where

$$T(\mu) = \frac{N\mu\sqrt{1.5}}{M\mu(1.5)/\sqrt{b} - N\sqrt{1.5}}.$$  

(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 $M\mu RTT_H > 4N$. Actually, for very low RTTs (4) is nonsensical; notice that if $M\mu RTT_H < 4\sqrt{N}$, the arrival rate $T(\mu)$ 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 $\lambda$ in terms of its input: $P_D$ and the system parameters $RTT$, $N$, $M$ and $\mu$. The open loop queuing model to be derived in Section III will find $P_D$ in terms of $\lambda$ and the system parameters $N$, $M$, $\mu$, $B$ 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 $\lambda$, $P_D$ 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 $\lambda$ along with parameters $\mu$, $M$, $K$ 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 Enset model in which there is an exponentially distributed delay with mean \(1/\lambda\) 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 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 Ensent 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 Ensent 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

\(^{1}\)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).
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, \( \tau \) 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, \( P_D/\mu \), 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

\[
\frac{1}{\lambda^*} = \frac{1}{\lambda^*(\lambda, P_D)} = \frac{1}{\lambda} + \frac{P_D}{\mu}.
\]

(5)

As shown in Appendix I, this yields a model whose mean arrival rate is equal to \( \Lambda \), 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

\[
\begin{align*}
\frac{p_{i+1}}{p_i} &= \begin{cases} 
(M - i) \frac{\lambda^*}{(i + 1)\mu} & 0 \leq i \leq K - 1 \\
(M - K) \frac{\lambda^*}{K\mu} & K \leq i \leq K + B - 1
\end{cases}
\end{align*}
\]

(6)

together with the normalization equation

\[
\sum_{i=0}^{K+B} p_i = 1.
\]

(7)

Recall that the mean arrival rate of the PEB model is equal to \( \Lambda \), 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

\[
P_D = \frac{(M - K)\lambda^* p_{K+B}}{\Lambda}.
\]

(8)

Notice that the steady state \( p_{K+B} \) is a function of \( \lambda^* \) which is actually determined by \( P_D \). This gives rise to another fixed-point relationship. If \( \lambda \) 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

\[
\lambda^* \approx \lambda \quad \text{for all } 0 \leq i \leq K + B
\]

into (6), and after using (6) and (7) to obtain \( \pi_{K+B} \), \( P_{D}(EB) \) is obtained by

\[
P_{D}(EB) = \frac{(M - K)\lambda p_{K+B}}{\Lambda_{EB}},
\]

(9)

where

\[
\Lambda_{EB} = \frac{M}{1/(\lambda + (1 - P_{D}(EB))/\mu)},
\]

(10)

which gives

\[
P_{D}(EB) = \frac{(M - K)p_{K+B}(\mu + \lambda)}{\mu + \lambda(M - K)p_{K+B}},
\]

(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 \( \lambda^* \) 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

<table>
<thead>
<tr>
<th>Event and corresponding switch operation</th>
<th>Corresponding PEB model approximation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. A packet arrives when a free port is available. It is then progressively directed to the output port.</td>
<td>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.</td>
</tr>
<tr>
<td>2. A packet arrives when all output ports and buffer spaces are busy. The packet is progressively discarded from the moment it arrives.</td>
<td>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).</td>
</tr>
<tr>
<td>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.</td>
<td>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.</td>
</tr>
<tr>
<td>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 ( \tau ). From time ( \tau ) before the packet finishes being transmitted, the packet does not occupy the input port, and new packets may arrive.</td>
<td>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 ). 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.</td>
</tr>
<tr>
<td>5. An output port becomes free and the buffer is empty. The number of busy output ports decreases by one.</td>
<td></td>
</tr>
</tbody>
</table>

The numerical results for the packet loss probability of the M/M/1/(K+B) queue in this paper were calculated using the closed form

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

where, using (3),

\[
A = \frac{\Lambda}{K \mu} = \frac{M \lambda}{K(\lambda + \mu)}. \tag{13}
\]

Recall that \( \lambda \) here is the reciprocal of the mean idle time of an input port, rather than the arrival rate which is its common meaning in queuing 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 \( M, B, K, \lambda \) and \( P_D \). One interesting regime is that in which \( K \) and \( M \) become large with \( \alpha \equiv K/M, \lambda \) and \( \mu \) fixed, and

\[
\hat{\rho} \equiv \lambda/(\lambda + \mu) < \alpha. \tag{14}
\]

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

\[
P_D \sim \left( \frac{(1-\alpha)^{B+1}}{\alpha^B} \right) \left( \frac{M}{K} \right)^{\left( \left( \frac{\lambda}{\mu} \right) + (1-\alpha) \right)} \left( \frac{\lambda(1-\alpha)}{\mu\alpha} \right)^B e^{-MD_{KL}(b(\alpha)||b(\hat{p}))} \sqrt{M},
\]

or more explicitly by

\[
P_D \sim \sqrt{\frac{1-\alpha}{2\pi\alpha}} \left( \frac{\lambda + \mu}{\mu} \right) \left( \frac{\lambda(1-\alpha)}{\mu\alpha} \right)^B \frac{e^{-MD_{KL}(b(\alpha)||b(\hat{p}))}}{\sqrt{M}},
\]

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

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

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),

\[
\rho C_{O} \approx R_{agg} = N \frac{\sqrt{1.5/P_D}}{RTT_H},
\]

where \( \rho \) is the bottleneck utilization, and \( C_O \) is the total output bit rate.

The relationship between bottleneck utilization and packet loss probability is thus

\[
\rho \approx \frac{\sqrt{1.5/P_D}}{RTT_H C_O/N}.
\]

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 bit-rate-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 \( RTT_H \), the only functional dependence that needs to be extrapolated is 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 \( \rho \) 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 Gbit/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 \( RTT_H = (180 - 60)/\log_e(180/60) = 109.1 \) 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
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/1/(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/1/(K+B) model, while not as accurate, is also quite close. In fact, when $B = 5$, the M/M/1/(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

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

Models PEB, WZ and M/M/1/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 $\lambda$ when the packet loss probability is high. Letting $x = K\mu/[M(1-K)]$, 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
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 4000\( K \). 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
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 \( M = 4K = 80 \)
Fig. 9. Bottleneck utilization vs. number of links per trunk ($K$).

(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.
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 \( \lambda_o \) be the arrival rate at the switch under the PEB model. Let

\[
\lambda_r = \lambda_o (1 - P_D)
\]

be the corresponding rate at which arrivals are accepted by the switch. Now \( \lambda_r / \mu \) is the carried load; by Little’s law \([40]\), \( \lambda_r / \mu \) is the mean number of busy output wavelengths giving

\[
\lambda_r = \mu \left( \sum_{i=0}^{K-1} ip_i + \sum_{i=K}^{K+B} Kp_i \right).
\]

The arrival rate \( \lambda_o \) is the probability-weighted average of the state-dependent arrival rates, namely

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

Rearranging and substituting (21) gives

\[
\lambda_o = \lambda^* M - \lambda^* \left( \sum_{i=0}^{K-1} ip_i + \sum_{i=K}^{K+B} Kp_i \right)
\]

\[
= \lambda^* (M - \lambda_r / \mu).
\]

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

\[
\lambda_o = \frac{M}{1 / \lambda^* + (1 - P_D) / \mu}
\]

\[
= \frac{M}{\lambda^{-1} + \mu^{-1}}
\]

which is the actual arrival rate \( \lambda \), 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 \( TAP_D / \mu \) to each of the \( M \) input links, reducing the number of arrivals by \( TAP_D \lambda / \mu \). (The losses add idle time of \( 1/\mu \) for each of the \( TAP_D \) loss events, while (5) adds idle time \( P_D / \mu \) for each of the \( TA \) 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 \( \lambda \) to \( P_\lambda(\lambda) \).
Making the dependence of \( p_i \) on \( \lambda^* \) explicit in (8), the
average packet loss probability is given by
\[
P_D = G(\lambda^*) \frac{(M - K)p_{K+B}(\lambda^*)}{\sum_{i=0}^{K+B} [M - \min(i, K)] p_i(\lambda^*)},
\]
(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 \( 1 \leq i \leq K + B \), (6) gives
\[
p_i(\lambda^*) = \frac{[M - \min(i - 1, K)] \lambda^*}{\min(i, K) \mu} p_{i-1}(\lambda^*)
\]
\[
= \left( \prod_{k=i}^{K+B} \alpha_k \right) (\lambda^*)(\mu)^i p_0(\lambda^*),
\]
where \( \alpha_i \equiv [M - \min(i - 1, K)] / \min(i, K) \mu > 0 \) does not depend
on \( \lambda^* \). We write \( \alpha_i = [M - \min(i, K)] \left( \prod_{k=i}^{K+B} \alpha_k \right) \)
cancel \( p_0(\lambda^*) \) in the numerator and denominator to obtain
\[
G(\lambda^*) = \frac{a_{K+B} \lambda^{K+B}}{\sum_{i=0}^{K+B} a_i \lambda^i} \frac{1}{1 + \sum_{i=0}^{K+B} (a_i/a_{K+B})(\lambda^*)^{i-K-B}}.
\]
(26)
Since \( a_i/a_{K+B} > 0 \) for all \( i \), the denominator is strictly
positive. It is clear that the denominator is decreasing in \( \lambda^* \)
and so \( G(\lambda^*) \) is increasing.

We observe that equations (5) to (8) provide an implicit
functional form for \( P_D \) in terms of \( \lambda^* \). Substituting \( P_D = G(\lambda^*) \)
(25) into
\[
\frac{1}{\lambda^*} = \frac{1}{\lambda} + \frac{P_D}{\mu},
\]
we can define \( H : (0, \infty) \to \mathbb{R} \) such that
\[
\frac{1}{\lambda} = H(\lambda^*) \triangleq \frac{1}{\lambda^*} - \frac{G(\lambda^*)}{\mu}.
\]
(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
\[
\lambda^* = H^{-1} \left( \frac{1}{\lambda} \right).
\]
(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
\[
P_D = P_\lambda(\lambda) = G(H^{-1}(1/\lambda)).
\]
(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
\[
P_\lambda(\infty) = G(H^{-1}(0)).
\]
(31)

**Theorem 1:** Let \( \tilde{P} = 1.5(N/(M\mu RTT))^2 \). If
\[
P_\lambda(\infty) > \tilde{P}
\]
then there exists a unique solution \( P_D^t \) to the closed-loop fixed point equation \( P_D = P_\lambda(T(P_D)) \) with \( T(P_D) \geq 0 \), and this solution satisfies
\[
\tilde{P} < P_D^t < P_\lambda(\infty)
\]
(33)
Otherwise, there is no solution with \( T(P_D) \geq 0 \).

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

Conversely, if (32) holds then there is a \( \lambda \) such that \( P_\lambda(\lambda) = \tilde{P} \). Analogously to (4), define \( f : (0, \infty) \to \mathbb{R} \) by
\[
f(\lambda) = \frac{N\mu\sqrt{1.5}}{M\mu RTT\sqrt{P_\lambda(\lambda)} - N\sqrt{1.5}} - \lambda.
\]
(34)
It remains to prove that the existence of \( \lambda \) implies there is
a unique solution to \( f(\lambda) = 0 \) with \( \lambda > 0 \). Note that \( f(\lambda) \) is
continuous for \( \lambda > 0 \) except at \( \lambda \) where the denominator
vanishes. Clearly, \( f(\lambda) < 0 \) for \( \lambda \in [0, \lambda] \), and so there is
no solution to \( f(\lambda) = 0 \) in that range. For \( \lambda > \lambda \), \( f(\lambda) \) is
well-defined, continuous and strictly decreasing because of the
properties of \( P_\lambda(\lambda) \). Moreover, \( f(\lambda) \to +\infty \) as \( \lambda \) decreases
to \( \lambda \). On the other hand, when \( \lambda \) becomes very large, \( f(\lambda) \)
becomes negative. To see this, note that the first part of it
is always positive and finite for \( \lambda > \lambda \) and decreasing in \( \lambda \).
Thus the second term \(-\lambda\) dominates. The intermediate value
theorem now guarantees a solution. The fact that \( f \) is strictly
decreasing in the interval \((\lambda, +\infty)\) now yields uniqueness.

It is possible to consider other discard models, \( P_\lambda \). 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 \( P_\lambda \) 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 \( \epsilon \), provided that each
iteration finds an approximation \( \Gamma(p) \) 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,
\[
P_D^t \in [p^- - \epsilon/2, p^+ + \epsilon/2]
\]
(35)
after each iteration. The base case follows from Theorem 1.
Consider without loss of generality the case the \( p < P_D^t \).
Algorithm 1 Calculate solution of \( p = \Gamma(p) \) to within \( \epsilon \)

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

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

\[
p_{K+B}(\lambda^*) = \frac{(M)_\lambda^{(M)} \lambda^{(K)}}{S_1(\lambda^*) + S_2(\lambda^*)}.
\]

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 \( \lambda \) and dividing numerator and denominator by \( (1 + \lambda/M)^M \) gives, after some algebra,

\[
p_{K+B}(\lambda^*) = \frac{(M)_\lambda^{(M)}(\hat{p})^{B-M-K}(\lambda \hat{p})^B}{D_1(\lambda) + D_2(\lambda)}
\]

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

\[
D_1(\lambda) = \frac{K}{\lambda^{(K)}} (\hat{p})^{K} (1 - \hat{p})^{M-K}(\lambda \hat{p})^B
\]

and

\[
D_2(\lambda) = \frac{K}{\lambda^{(K)}} (\hat{p})^{K} (1 - \hat{p})^{M-K}(\lambda \hat{p})^B \frac{1 - \lambda \hat{p}}{1 - \lambda \hat{p}}.
\]

Lemma 1: Consider the limit as \( K \rightarrow \infty \) and \( M \rightarrow \infty \) with \( \alpha = K/M, \lambda \) and \( \mu \) fixed, and \( \hat{p} < \alpha \). Then

\[
p_{K+B}(\lambda^*) \sim \frac{(M)_\lambda^{(M)}(\hat{p})^{K} (1 - \hat{p})^{M-K}(\lambda \hat{p})^B}{\lambda^{(K)}} \frac{1 - \lambda \hat{p}}{1 - \lambda \hat{p}}.
\]

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 \( \alpha \), and \( \hat{p} < \alpha \). By Tchebycheff’s inequality

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

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

\[
\frac{(M)_\lambda^{(M)}(\hat{p})^{M-K}}{\hat{p}^{K} (1 - \hat{p})^{M-K}}.
\]

where \( K = \alpha M \) and \( \hat{p} < \alpha \). We compare (43) with \( (M)_\lambda^{(M)}(\hat{p})^{M-K} \), 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

\[
\frac{\alpha^K(1 - \alpha)^{M-K}}{\hat{p}^{K} (1 - \hat{p})^{M-K}}.
\]
We rewrite this as
\[
\left( \frac{\alpha^\alpha(1-\alpha)^{1-\alpha}}{(\hat{p}^\alpha(1-\hat{p})^{1-\alpha})} \right)^M
\]
(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
\[
\alpha \left[ \log \left( \frac{\alpha}{\alpha-1} + (1-\alpha) \right) \right] (1-\alpha) - \log \left( 1 - (1 - \hat{p}) \right). \tag{46}
\]

This is the Kullback-Leibler divergence \( D_{KL}(b(\alpha) || b(\hat{p})) \) 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 \( D_{KL}(b(\alpha) || b(\hat{p})) \) becomes zero for \( \hat{p} = \alpha \). This gives that (44) is asymptotically \( e^{-MD_{KL}(b(\alpha) || b(\hat{p}))} \), and so that
\[
\left( \frac{M}{K} \right)^{\hat{p}^K(1-\hat{p})^{M-K}} \sim \frac{1}{\sqrt{2\pi M \alpha (1-\alpha)}} e^{-MD_{KL}(b(\alpha) || b(\hat{p}))}.
\tag{47}
\]

As a result
\[
P_{K+B}(\lambda) \sim (V\lambda)^B \frac{1}{\sqrt{2\pi M \alpha (1-\alpha)}} e^{-MD_{KL}(b(\alpha) || b(\hat{p}))}.
\tag{48}
\]

By (8) and (3), we have
\[
P_D = \frac{(M-K)(\lambda + \mu)\lambda^\alpha}{M\mu} P_{K+B}(\lambda^*),
\tag{49}
\]
so that
\[
P_D < C_1(1-\alpha) \frac{(\lambda + \mu)\lambda^\alpha}{\lambda^\mu} (V\lambda)^B \frac{1}{\sqrt{2\pi M \alpha (1-\alpha)}} e^{-MD_{KL}(b(\alpha) || b(\hat{p}))},
\tag{50}
\]

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 \( \hat{p}^* = \lambda^* / (\lambda^* + \mu) \),
\[
\frac{1}{\hat{p}^*} - 1 = \frac{1}{\lambda} - \frac{P_D}{\mu} \lambda,
\tag{52}
\]
\[
P_D < C_2 \frac{1}{\sqrt{M}} e^{-MD_{KL}(b(\alpha) || b(\hat{p}))},
\tag{53}
\]

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

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
\[
\left( \frac{1}{\hat{p}^*} \right)^M \left( \frac{1}{\hat{p}} \right)^{M-K} \left( \frac{\lambda^*}{\lambda} \right)^B.
\tag{55}
\]

By equation (52), this can be expressed as
\[
(1 - \hat{p}^*)^M \left( \frac{1}{\hat{p}} \right)^{M-K}.
\tag{56}
\]

Taking log of each factor, using \( \log(1+x) \leq x \) and substituting (52) yields
\[
M \log(1 - \hat{p}^*) P_D \geq -CM e^{-MD_{KL}(b(\alpha) || b(\hat{p}))},
\tag{57}
\]
and
\[
M(1-\alpha) \log \left( 1 + \frac{P_D}{(1-\hat{p})} \right) \leq C' e^{-MD_{KL}(b(\alpha) || b(\hat{p}))}.
\tag{58}
\]

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).

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

**References**


http://www.isi.edu/nsnam/ns/.


Proc. IEEE IN-...
Lachlan L. H. Andrew (M’97–SM’05) received the B.Sc. degree in computer science in 1992, the B.E. degree in electrical engineering in 1993, and the Ph.D. degree in engineering in 1997, all from the University of Melbourne, Australia.

Since 2008, he has been an Associate Professor at Swinburne University of Technology, Australia. From 2005 to 2008, he was a senior research engineer in the Department of Computer Science at Caltech. Prior to that, he was a senior research fellow at the University of Melbourne and a lecturer at RMIT, Australia. His research interests include performance analysis of congestion control, resource allocation algorithms and energy-efficient networking. He was co-recipient of the best paper award at IEEE MASS’07.

Dr. Andrew is a member of the IET and the ACM.

Tony Cui received the B.Sc. degree in Electronics Engineering from Tsinghua University at Beijing, P. R. China in 1997 and the Ph.D. degree from the Melbourne University in 2008. His research interest includes protocol performance analysis and congestion control in the Internet.

Dr. Cui won of Australian Academy of Technological Sciences and Engineering Fellowship for young scientists and engineers (ATSE ECFSE) in 2006.

Bill Moran (M’95) Bill Moran is the Research Director of Melbourne Systems Laboratory (MSL); a research institute in the Department of Electrical and Electronic Engineering at the University of Melbourne, Australia, where he has been a Professor of Electrical Engineering since 2001. Previously he was Professor of Mathematics (’76–’91), Head of the Department of Pure Mathematics (’77–’79, ’84–’86), Dean of Mathematical and Computer Sciences (’81, ’82, ’89) at the University of Adelaide, and Head of the Mathematics Discipline at the Flinders University of South Australia (’91–’95). He was a Chief Investigator (’92–’95), and Head of the Medical Signal Processing Program (’95–’99) in the Cooperative Research Centre for Sensor Signal and Information Processing. He was elected to the Fellowship of the Australian Academy of Science in 1984. He holds a Ph.D. in Pure Mathematics from the University of Sheffield, UK (’68), and a First Class Honours B.Sc. in Mathematics from the University of Birmingham (’65).

He has been a Principal Investigator on numerous research grants and contracts, in areas spanning pure mathematics to radar development, from both Australian and US Research Funding Agencies, including DARPA, AFOSR, AFRL, Australian Research Council (ARC), Australian Department of Education, Science and Training, DSTO. He is a member of the Australian Research Council College of Experts.

His main areas of research interest are in signal processing both theoretically and in applications to radar, waveform design and radar theory, sensor networks, and sensor management. He also works in various areas of mathematics including harmonic analysis, representation theory, and number theory.

Andrew Zalesky received his B.Sc. in 2002, B.E. in electrical engineering in 2003 and Ph.D. in engineering in 2006, all from the University of Melbourne, Australia. In 2008, he was an Australian Research Council (ARC) International Fellow. He is currently an ARC Postdoctoral Fellow based at the University of Melbourne. His research interests are in modeling and performance evaluation of natural and engineered systems. Since commencing postdoctoral research, he has received support from the ARC, Australian Academy of Science, American Australian Association and the CASS Foundation.

Moshe Zukerman (M’87–SM’91–F’07) received the B.Sc. degree in industrial engineering and management and the M.Sc. degree in operations research from TechnionIsrael Institute of Technology, Haifa, Israel, and the Ph.D. degree in electrical engineering from University of California, Los Angeles, in 1985. He was an independent Consultant with the IRI Corporation and a Postdoctoral Fellow with the University of California, Los Angeles, in 1985–1986. In 1986–1997, he was with the Telstra Research Laboratories (TRL), first as a Research Engineer and, in 1988–1997, as a Project Leader. He also taught and supervised graduate students at Monash University in 1990–2001. Between 1997-2008, he was with The University of Melbourne, Victoria, Australia, promoting and expanding telecommunications research and teaching in the Electrical and Electronic Engineering Department. In 2008 he joined the Electronic Engineering Department, City University of Hong Kong, where he is now a Chair Professor of Information Engineering and a project leader. He has over 200 publications in scientific journals and conference proceedings. He has served on various the editorial boards such as Computer Networks, IEEE Communications Magazine, IEEE Journal of Selected Ares in Communications, IEEE Transactions on Networking and the International Journal of Communication Systems.