

# Computing Path Blocking Probability and Delay in Optical Networks With Retrial

Onur Alparslan, Shin'ichi Arakawa, and Masayuki Murata

**Abstract**—Hybrid optical architectures combining path and packet switching can be good candidates for future optical networks because they exploit the best of both worlds. However, the optimization of some parameters of the hybrid switch by some metrics is vital to maximize the benefit of the hybrid architecture. Blocking rate and reservation delay are two of the most important performance metrics in the path switching layer of the hybrid architecture. In this paper, we propose an analytical method with an improved backward blocking analysis to compute both blocking rates and reservation delays in path switching optical WDM networks with destination-initiated reservation allowing retrial of failed reservation attempts. On a mesh topology, we show that the results of our analytical method and simulations were close to each other, while the analytical method was several orders of magnitude faster than the simulation, which may allow much faster performance analysis, design, and optimization of hybrid networks.

**Index Terms**—Analytical model; Blocking probability; Path switching; Reservation delay; Retrial; Wavelength-division multiplexing.

## I. INTRODUCTION

Optical fiber with wavelength division multiplexing (WDM) allows much higher bandwidth and can span longer distances than electrical cabling, so it is a promising solution to handle the fast-growing Internet traffic that is demanding more and more capacity. WDM can use different switching granularities in order to utilize the vast capacity of fiber links, e.g., packet, burst, and path (circuit) switching, where each of them have pros and cons. While optical packet switching allows higher utilization of WDM channels thanks to its high statistical multiplexing gain and flexibility, it has disadvantages like higher switch cost as it needs ultrafast switching fabric to achieve high granularity. Moreover, the current optical buffering technology is not mature enough to provide large and fast buffering space to optical packet switching. On the other hand, path switching has many advantages over packet switching like low switch cost and power requirements as its switching speed and frequency are lower. Moreover, it does not need optical buffering at the core nodes as there is no contention

of packets, so it has an easier and more effective quality of service (QoS) support for flows with strict QoS requirements. However, path switching has lower utilization efficiency because a connection may or may not use all the capacity in the dedicated channel. Moreover, path switching needs prior reservation of channels, which adds an additional delay to flow completion time.

A hybrid architecture combining path and packet switching is a possible solution to these problems by exploiting the best of both worlds [1,2]. There are two main approaches in the literature for realizing a hybrid architecture. One of them is carrying both packet and path traffic on the same wavelength [3,4]. All wavelengths are principally used by paths. The packet traffic is inserted into idle periods left from the path traffic on the same wavelength. Another method is to use separate wavelengths for path and packet switching, and distribute the traffic between them. For example, the network can carry short flows over packet switching wavelengths while carrying the large flows on path switching wavelengths [5]. Both methods need optimization of some parameters of the hybrid switch in order to minimize the flow completion time while keeping the hardware cost low. For example, the switching capacity of packet switching fabric in both methods and the optimum ratio of path and packet-switching wavelengths in the second method should be optimized. Optimization of these parameters requires fast and easy calculation of some performance metrics for path and packet-switched networks.

The key performance metrics in path-switched networks are the blocking probability and the reservation delay, which is the time spent for the reservation. The choice of the wavelength reservation algorithm has a big impact on both metrics. One of the most popular reservation algorithms in the literature is destination-initiated reservation (DIR) [6]. Resource reservation protocol-traffic engineering (RSVP-TE) [7] signaling protocol in Generalized Multi-Protocol Label Switching (GMPLS) [8] networks uses DIR for wavelength reservation. In DIR, when there is a connection request, the source node sends a PROBE packet, which collects a list of idle wavelengths along the path. The destination node selects one of the wavelengths, which is idle on all links in order to satisfy the wavelength-continuity constraint [9] when there is no wavelength conversion ability in the network. In case there is no idle wavelength left in the list, the node sends a P\_NACK packet to the source, which causes the connection request to be dropped at the source, and this is called

Manuscript received December 7, 2012; revised April 2, 2013; accepted April 2, 2013; published April 29, 2013 (Doc. ID 181313).

The authors are with the Graduate School of Information Science and Technology, Osaka University, 1-5 Yamadaoka, Suita, Osaka 565-0871, Japan (e-mail: a-onur@ist.osaka-u.ac.jp).

<http://dx.doi.org/10.1364/JOCN.5.000498>

forward blocking. If the destination selects an idle wavelength, it sends a RESV packet to the source node in order to reserve it along the path. However, a previously idle wavelength may have been reserved by another connection when the reservation packet arrives. This is called backward blocking. In this case, the RESV packet is converted to an R\_NACK packet, and reservation is no longer done in the rest of the path. If the source node receives an R\_NACK packet, it again drops the connection request and sends a RELEASE packet to the destination to release the reservations done by the RESV packet. A RELEASE packet may also be sent from the failed node for faster release instead of the source node, but in this work, we use the conservative method, in which a RELEASE packet is sent by source nodes [10]. If the source node receives a RESV packet, it means that the selected wavelength has been reserved successfully along the path, so it sends the data over this wavelength. When the flow is finished, the source node sends a RELEASE packet to remove the reservation of the reserved wavelength.

Several analytical models for calculating the forward and backward blocking rates in path switching have been proposed in the literature. Most of them are based on the reduced load approximation (RLA) method, which calculates the blocking rates in an iterative manner [11]. The initial analytical models in the literature were on calculating only the forward blocking caused by an insufficient number of channels to accept all the reservation requests. In [12], the forward blocking rate is calculated by using the RLA method and considering the state-dependent arrival rate of flows by solving an  $M/M/c/c$  birth-death process. However, the analysis in [12] is for electronic circuit-switching networks, so it does not take the wavelength-continuity constraint into account. The wavelength-continuity constraint is introduced in [13] and [14], where Birman's method [13] is more advanced as it includes the state-dependent arrival of flows like in [12]. Computational complexity in Birman's method increases with the path length, so a different model based on the inclusion-exclusion principle was proposed in [15] to lower the computation complexity independent of the path length. Moreover, a link correlation model was proposed to get more precise results on sparse networks. Finally, in [16], an original method is proposed that analyzes the network by decomposing it into single-path subsystems and constructing an exact Markov process that captures the behavior of a path in terms of wavelength use.

To the best of our knowledge, the first analytical model that includes the calculation of the backward blocking rate was proposed in [10]. The backward blocking rate is calculated by incorporating the wavelength reservation duration and propagation delays in the analysis to include blocking due to outdated information. It calculates the forward blocking rate by considering each wavelength as an  $M/G/1/1$  queuing model to obtain the stationary probability of each wavelength. However, this method does not take the state-dependent arrival rate of flows into account, so it has a higher error rate at high traffic load when compared with [13]. Another analysis that includes the backward-blocking is in [17], which calculates the forward-blocking

by using Birman's method [13], so the forward blocking calculation is more precise than that of [10]. However, the backward blocking analysis in [17] makes too many simplifying assumptions, which make it less precise than the backward blocking analysis in [10]. Reference [18] improves the model by applying the inclusion-exclusion principle, which was proposed in [15].

While the forward blocking analysis based on the inclusion-exclusion method like [15] and [18] have a lower computation complexity than does Birman's method, the processing speed of a modern CPU is enough to solve large topologies with ease using Birman's method. Unlike the inclusion-exclusion method, Birman's method does not take load correlation introduced by the wavelength continuity constraint into account, so it tends to overestimate the blocking in sparsely connected networks with high traffic load. However, its impact was found to be small as we will show in this paper. On the other hand, [15] states that the combinatorial term in the model of the inclusion-exclusion method becomes extremely huge when the wavelength count is higher than 64 and introduces significant round-off errors if the blocking probabilities are small ( $\approx 10^{-3}$ ) so it must be used with caution when analyzing networks with many wavelengths and low blocking probabilities. However, commercial networks with 160 wavelengths per fiber using dense wavelength division multiplexing are already available and being used, so the inclusion-exclusion method may not be suitable for modeling today's high capacity networks. Therefore, we chose and applied Birman's method for the calculation of forward blocking in our analysis. However, both forward blocking methods use a similar queuing analysis framework and they are modular, so it is also possible to use our proposed backward blocking analysis with the forward blocking calculation by the inclusion-exclusion method for calculating the blocking with high precision in sparsely connected networks with high load correlation.

As shown in [18], forward blocking occurs due to insufficient network capacity. Many network operators have run their networks at low utilization traditionally to keep the blocking rates low, so most of the blocking takes place in the backward direction [18]. Recently, more WDM networks are modestly or heavily utilized because the traffic volume may increase faster than the network capacity due to costs or the technical limitations. However, even in highly congested networks a high proportion of the blocking is due to the backward blocking as we will show in the numerical results section. Therefore, the accuracy of the backward blocking analysis has a great impact on the accuracy of the overall blocking calculation. In this paper, which is an extended version of [19], we propose a new analytical method, which greatly improves the backward blocking analysis in [10] by introducing estimation of the state-dependent arrival rate of RESV packets for the backward reservation for more precise results and adapts it for use with Birman's forward blocking analysis for an iterative calculation. As we show in this paper, the analytical modeling with our improved backward blocking analysis coupled with Birman's forward blocking analysis allows fast and highly accurate calculation of overall blocking over a wide range of load and topology scenarios.

In [19], we calculated the blocking rates on the National Science Foundation Network (NSFNET) topology. In this paper, we switched to the European Optical Network (EON) topology, which is newer and larger than NSFNET. Moreover, we extended the analytical model to calculate the reservation delays for estimating the flow completion time in the path network. If the size of a new coming flow is known, we can estimate its completion time in path and packet switching layers and assign the flow to the layer with minimum completion time. Furthermore, we extended the analytical model to incorporate the retrial of failed reservation attempts. Retrial can greatly decrease the blocking rates and, hence, increase the efficiency of the path network, thus decreasing the need for packet wavelengths in a hybrid architecture. However, retrial may also increase the flow completion times, so the maximum number of retrials and the back-off delay time should be selected carefully. To the best of our knowledge, an analysis with connection request retrials was first presented in [20], which assumes the blocked connections retry according to a probability distribution. However, the probabilistic model in [20] can limit the maximum number of retrials to only zero and one. If more than one retrial is allowed, no strict limit can be imposed on the maximum number of retrials, so a flow may end up retrying too many times. In our model, we can limit the maximum number of retrials to any value, as well as set a retrial probability, so we can prevent a flow from making too many retrials while controlling the retrial probability.

The paper is organized as follows. In Section II, we propose an analytical model for calculating the blocking probability and reservation delay of the DIR method with a retrial option. Section III compares the analytical and simulation results and discusses the effects of retrials and back-off delay on network performance. Section IV concludes the paper.

## II. ANALYTICAL MODEL

The blocking in DIR can be in the forward or backward direction. The forward blocking occurs when there is not a wavelength that is idle on all links along the path, while the backward blocking occurs when a previously idle and selected wavelength is reserved by another connection before the returning reservation packet arrives. The forward blocking is caused by insufficient network capacity, while the backward blocking is caused by outdated information [18].

### A. Forward Blocking Analysis

We use Birman's method to calculate the forwarding blocking rate [13]. We consider an arbitrary topology consisting of  $J$  links and  $W$  wavelengths. When a PROBE packet arrives to its destination node, the DIR algorithm randomly selects one of the wavelengths that satisfies the wavelength-continuity constraint. There is a fixed route between each node pair. We assume Poisson flow arrival, which is shown to hold on real core networks where

a large number of flows are multiplexed [21]. Then, the number of busy wavelengths on a link can be modeled by a birth-death process ( $M/G/c/c$  queueing system, also known as the Erlang loss model), as shown in Fig. 1. The state (wavelength occupancy) probabilities can be calculated by the well-known Erlang equations:

$$p_{m,k}^n = \left[ \frac{\alpha_{m,0}^n \alpha_{m,1}^n \cdots \alpha_{m,k-1}^n}{k! (\mu_m^n)^k} \right] p_{m,0}^n \quad (1)$$

and

$$p_{m,0}^n = \frac{1}{1 + \sum_{z=1}^W \frac{1}{z!} \prod_{j=0}^{z-1} \frac{\alpha_{m,j}^n}{\mu_m^n}}, \quad (2)$$

where

- $k$  denotes the number of busy wavelengths.
- $p_{m,k}^n$  denotes the wavelength occupancy probability of having exactly  $k$  busy wavelengths on the  $n$ th link of node pair  $m$ .
- $\alpha_{m,k}^n$  denotes the total reservation arrival rate when there are exactly  $k$  busy wavelengths on the  $n$ th link of node pair  $m$ .
- $\mu_m^n$  denotes the reservation departure rate on the  $n$ th link of node pair  $m$ . The Erlang loss model is insensitive to connection holding time distribution, so  $\mu_m^n$  can be any distribution.

Let  $q_{m,h}^n$  be the probability that  $h$  wavelengths do not satisfy the wavelength continuity constraint along the first  $n$  hops of a node pair  $m$  with a total hop length of  $d$ , as they are busy on at least one of the hops. For the first hop, the probability is simply  $q_{m,h}^1 = p_{m,h}^1$ . If we assume a mutual independence of wavelength distribution between adjacent links, on the second hop of a path, we can write

$$q_{m,h}^2 = \sum_{i=0}^W \sum_{j=0}^W R(W-h|W-i, W-j) p_{m,i}^1 p_{m,j}^2, \quad (3)$$

where

$$R(h|i,j) = \frac{\binom{i}{h} \binom{W-i}{j-h}}{\binom{W}{j}}, \quad (4)$$

if  $\max(0, i+j-W) \leq h \leq \min(i,j)$  and is equal to 0 otherwise. Equation (4) is the conditional probability of having  $h$  wavelengths idle on both links, given that there are  $i$  idle

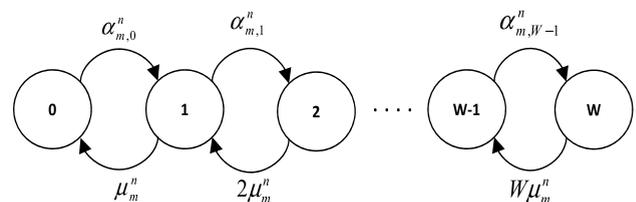


Fig. 1. Birth-death process.

wavelengths on the first link and  $j$  idle wavelengths on the second link. Equation (3) is the sum of the probability of all possible wavelength occupancy distributions on both links, weighted by the conditional probability of having  $h$  wavelengths that do not satisfy the wavelength continuity constraint. For the  $n$ th link of a path, we can extend Eq. (3) to calculate  $q_{m,h}^n$  recursively by

$$q_{m,h}^n = \sum_{i=0}^W \sum_{j=0}^W R(W-h|W-i, W-j) q_{m,i}^{n-1} p_{m,j}^n, \quad (5)$$

where  $p_{m,i}^1$  in Eq. (3) is replaced by  $q_{m,i}^{n-1}$ . When the wavelength count is high, calculating Eq. (3) may become the most time-consuming part of the algorithm. We solved it by precalculating a lookup table populated with the results of Eq. (3) for all possible  $i, j$ , and  $h$  values. As Eq. (3) is calculated with the same parameters many times due to iterations and recursions in the algorithm, the lookup table greatly improves the speed of the analytical calculation.

Let  $e_m$  be the departure rate of PROBE packets from the source node and  $\lambda_{m,k}^n$  be the arrival rate of PROBE packets from a node pair  $m$  to the  $n$ th link of its path that are not blocked on this link (satisfying the wavelength continuity constraint) when there are  $k$  busy wavelengths on the  $n$ th link. If the path has a single hop, the arrival rate of PROBE packets to the destination node simply equals  $\lambda_{m,k}^1 = e_m$ . In case of a multihop connection, the arrival probability of a reservation request to a link depends on blocking probabilities on previous links and the wavelength occupancy of the selected link. In case of a two-hop path, the arrival probability can be calculated similar to Eq. (3). We know the number of busy wavelengths on the second link, so  $p_{m,j}^2$  becomes 1 and the inner sum is eliminated in Eq. (3). Reservation requests arrive as long as there is a wavelength satisfying the wavelength continuity constraint, so  $\lambda_{m,k}^2$  for a two-hop path, when there are exactly  $k$  busy wavelengths on the second link, can be calculated by

$$\begin{aligned} \lambda_{m,k}^2 &= e_m \sum_{h=0}^{W-1} \sum_{i=0}^W R(W-h|W-i, W-k) p_{m,i}^1 \\ &= e_m \left( 1 - \sum_{i=0}^W R(0|W-i, W-k) p_{m,i}^1 \right). \end{aligned} \quad (6)$$

In case of an  $n$ -hop path, we can calculate  $\lambda_{m,k}^n$  recursively [like in Eq. (5)] by

$$\lambda_{m,k}^n = e_m \left( 1 - \sum_{i=0}^W R(0|W-i, W-k) q_{m,i}^{n-1} \right). \quad (7)$$

Let  $\gamma_m^n$  be the average rate of call setup requests from node pair  $m$  that reserves a wavelength successfully on the  $n$ th link of its path. On the last link  $d$  of a node pair  $m$ , it is calculated by

$$\gamma_m^d = e_m \sum_{h=0}^{W-1} q_{m,h}^d = e_m (1 - q_{m,W}^d), \quad (8)$$

where  $q_{m,W}^d$  [calculated by Eq. (5)] is the forward blocking rate of the node pair, which means that there is no wavelength that satisfies the wavelength continuity constraint along the path. The successful call setup requests select an idle wavelength randomly and try to reserve the same wavelength number along the path from destination to source node. The reader is referred to [13] for more detail on forward blocking calculations.

## B. Backward Blocking Analysis

Next, we calculate the rate of backward reservation requests, which are categorized into two classes:

- (1) Class 1: The selected wavelength is available at all the links along the path, so it will be reserved and the data transmission will occur. Let  $\delta_m$  be the rate of class 1 traffic for node pair  $m$ .
- (2) Class 2: The selected wavelength has already been reserved at some upstream link by another node pair, so the reservation and the data transmission will fail. Let  $\beta_m^n$  be the rate of class 2 traffic for node pair  $m$  on the  $n$ th link of its path.

First, we need to derive the probability that a selected wavelength that was idle when the PROBE packet arrived is still not reserved by other interfering node pairs when the reserve packet arrives to that link on the backward path after some delay. For this purpose, we should know the reservation arrival rates of interfering node pairs. There may be two types of interfering reservation request arrivals on the  $n$ th link. The first type comes from the node pairs that will do their first reservation on this link because  $n$  is the last link on their path. The second type comes from the interfering node pairs that have the link  $n$  on their path, but  $n$  is not their last link. An important point is that if the path of two node pairs interferes at two or more links, backward reservation contention occurs only at the first interfering link  $n$ , which is the one closest to the destination. Therefore, there should be no interference at links  $n+c$ , where  $c > 0$ . The original backward blocking model in [10] does not take this into account, but we improved the model to handle this situation. Let  $\Lambda_{m,k}^n$  be the total arrival rate of the reservation requests of node pairs interfering with requests from node pair  $m$  when there are  $k$  busy wavelengths on link  $n$ .  $\Lambda_{m,k}^n$  can be calculated by

$$\Lambda_{m,k}^n = \sum_{\substack{m' \in M, \\ n' = m_n, \\ n' = d(m')}} \lambda_{m',k}^{n'} + \sum_{\substack{m' \in M, \\ n' \neq n, \\ m' = m_n, \\ n' \neq m_n + c, \\ n' \neq d(m')}} \gamma_{m',k}^{n'} \quad (9)$$

for the first  $d-1$  links of node pair  $m$ , where

- $\gamma_{m,k}^n$  denotes the rate of call setup requests from a node pair  $m$  that reserves a wavelength successfully on the  $n$ th link when there are  $k$  busy wavelengths on the link.

- $m_n$  denotes the link id of the  $n$ th link of node pair  $m$  in the overall topology.
- $M$  denotes the set of all node pairs in the network.
- $d(m)$  denotes the hop count of node pair  $m$ .

The first summation in Eq. (9) is the sum of the arrival rates  $\lambda_{m',k}^{n'}$  of PROBE packets from the node pairs  $m' \in M$  that will do reservation on this link, which is the last link on their path (expressed by  $n' = d(m')$ ), and it is the same as the  $n$ th link of the node pair  $m$  (expressed by  $m'_n = m_n$ ). Also the flows in the same node pair  $m$  compete for reservation when this is the last link of  $m$ , so  $m$  is included in  $M$ . The second summation is the sum of the call setup requests  $\gamma_{m',k}^{n'}$  by RESV packets from the node pairs  $m' \in M$ , whose  $n$ th link is the same as the  $n$ th link of the target node pair  $m$  (expressed by  $m'_n = m_n$ ), while this is not the last link of their path (expressed by  $n' \neq d(m')$ ) and reserves a wavelength successfully on the  $n$ th link and this is the last link that  $m$  and  $m'$  interfere (expressed by  $m'_{n'+c} \neq m_{n+c}$ ). The flows in the same node pair  $m$  do not interfere except on the last link of the path, so the traffic from  $m$  is excluded in the second summation (expressed by  $m' \neq m$ ). The value of  $\lambda_{m',k}^{n'}$  and  $\gamma_{m',k}^{n'}$  variables comes from the previous iteration of the algorithm.

The interfering traffic  $\Lambda_{m,k}^n$  causes backward blocking, which decreases the arrival rate of the reservation requests of a node pair  $m$  at each interfering hop on the way to the source node. Let  $D$  be the two-way propagation delay of a link. We show it as a constant to simplify the notations, but it is possible to calculate with different link delays in the network. Assuming that the interfering traffic arrival is Poisson, the probability that no other reservation request arrives to a selected wavelength in a time interval  $(t, t + \tau]$  can be calculated by  $e^{-\lambda\tau}$ , where  $\lambda$  is the arrival rate of interfering reservation requests per idle wavelength. The time interval between the PROBE packet of node pair  $m$  with a total hop length of  $d$  checks the wavelength availability on link  $n - 1$  and the returning RESV packet tries to reserve a wavelength on the same link and is  $\tau = (d - n + 1)D$ . The arrival rate of interfering reservation requests per idle wavelength for node pair  $m$  when there are  $k$  busy wavelengths on the link  $n - 1$  is  $\lambda = \Lambda_{m,k}^{n-1}/(W - k)$ . Therefore, we can estimate the arrival rate of reservation requests from node pair  $m$  that succeeds in reservation on link  $n - 1$  when there are  $k$  busy wavelengths by

$$\gamma_{m,k}^{n-1} = \gamma_m^n e^{-\Lambda_{m,k}^{n-1}(d-n+1)D/(W-k)}, \quad (10)$$

which will be used in Eq. (9) in the next iteration of the algorithm. Moreover, we calculate the overall reservation arrival rate from node pair  $m$  that succeeds in reservation on link  $n - 1$  by weighting the state (wavelength occupancy) dependent arrival rates in Eq. (9) with the state probabilities in Eq. (1) by

$$\gamma_m^{n-1} = \sum_{j=0}^{W-1} P_{m,j}^{n-1} \gamma_{m,j}^{n-1} = \gamma_m^n \sum_{j=0}^{W-1} P_{m,j}^{n-1} e^{-\Lambda_{m,j}^{n-1}(d-n+1)D/(W-j)}. \quad (11)$$

Reference [10] uses an expected wavelength occupancy ratio for calculating the reservation arrival rates. However,

our model improves the calculation of backward blocking by estimating a specific reservation arrival rate for all possible wavelength occupancy ratios by using Eqs. (9)–(11).

As a result of Eq. (11), class 1 traffic can be calculated for all links on the path recursively by

$$\delta_m = \gamma_m^1 = \gamma_m^d \prod_{x=2}^d \sum_{j=0}^{W-1} P_{m,j}^{x-1} e^{-\Lambda_{m,j}^{x-1}(d-x+1)D/(W-j)}. \quad (12)$$

The arrival rate of class 2 traffic on link  $n$  is simply the difference between the rate of reservation requests from a node pair  $m$ , which reserves a wavelength successfully on the link  $n$ , and the rate of class 1 traffic by

$$\beta_m^n = \gamma_m^n - \delta_m. \quad (13)$$

Let  $s_m^n$  and  $t_m^n$  be the mean occupation times for class 1 and 2 traffic on the  $n$ th link of the path of node pair  $m$ . Let  $\varphi$  be the mean occupation time of data transfer. If a reservation succeeds in all links along the path, the time interval between when a wavelength is reserved on link  $n$  and the data traffic arrives is  $nD$ , so the class 1 mean occupation time is

$$s_m^n = nD + \varphi. \quad (14)$$

If a reservation fails eventually, the time interval between when a wavelength is reserved on link  $n$  and a P\_NACK arrives is  $nD$  if  $n \geq 2$ . There is no class 2 traffic on the first link. Therefore, the class 2 mean occupation time is

$$t_m^n = \begin{cases} nD & \text{if } n \geq 2 \\ 0 & \text{otherwise} \end{cases}. \quad (15)$$

The mean departure rate, which is the inverse of the average wavelength occupation time, is calculated by

$$\mu_m^n = \frac{\sum_{\substack{m' \in M \\ m'_n = m_n}} \gamma_{m'}^{n'}}{\sum_{\substack{m' \in M \\ m'_n = m_n}} (\delta_{m'} s_{m'}^{n'} + \beta_{m'}^{n'} t_{m'}^{n'})}, \quad (16)$$

where the class 1 and 2 mean occupation times are weighted by their traffic rate for all  $m' \in M$ .

The overall reservation arrival rate when there are  $k$  busy wavelengths on the  $n$ th link of node pair  $m$  is

$$\alpha_{m,k}^n = \sum_{\substack{m' \in M \\ m'_n = m_n \\ n' = d(m')}} \lambda_{m',k}^{n'} + \sum_{\substack{m' \in M \\ m'_n = m_n \\ n' \neq d(m')}} \gamma_{m',k}^{n'}, \quad (17)$$

where the first summation is the sum of the arrival rates  $\lambda_{m',k}^{n'}$  of PROBE packets from the node pairs  $m' \in M$  that will select and reserve a wavelength successfully on this link, which is the last link on their path. The second summation is the sum of the call setup request rates  $\gamma_{m',k}^{n'}$  by RESV packets from the node pairs  $m' \in M$  that will do reservation on this link successfully, whereas this is not the last link on their path. Unlike in Eq. (9), the traffic from the node pair  $m$  is included in the second summation of

Eq. (17). We note that  $\mu_m^n$ ,  $\alpha_{m,k}^n$ , and  $p_{m,k}^n$  are the same for all  $m' \in M$  using the link, so it is enough to calculate them once per link.

Finally, using Eqs. (8) and (12), the blocking probability of a node pair  $m$  with hop count  $d$  is

$$L_m = 1 - \frac{\delta_m}{e_m} = 1 - (1 - q_{m,W}^d) \prod_{x=2}^d \sum_{j=0}^{W-1} P_{m,j}^{x-1} e^{-\Lambda_{m,j}^{x-1}(d-x+1)D/(W-j)}. \quad (18)$$

We used the following algorithm to calculate blocking probability by using the RLA method iteratively:

- (1) Initialize  $L_m$  for all the node pairs to zero. Initiate state dependent arrival rates as if there is no blocking in the network.
- (2) Calculate the wavelength occupation time  $\mu_m^n$ .
- (3) Calculate the state-dependent arrival rate  $\alpha_{m,k}^n$ .
- (4) Derive the new blocking probability  $L_m$ . If the difference between the old and new values of  $L_m$  for each node pair is less than a small constant (we used  $10^{-7}$  in this paper), then finish the iteration. Otherwise, return to step 2 and begin the next iteration.

### C. Retrial Analysis

Assume that an incoming call setup request attempts to reserve a connection up to  $l$  times until reservation succeeds. The trials that are blocked retry with a probability  $r$  or give up and leave the system with a probability of  $1 - r$ . The requests that are blocked  $l$  times are dropped. For the sake of simplicity, we assume that the network conditions at reservation attempts are uncorrelated. Therefore, each attempt has the same blocking probability. Let  $e_m^l$  be the total departure rate of PROBE packets from the source node of a node pair  $m$  with a limit of  $l$  reservation attempts ( $l - 1$  retrials). The  $e_m^l$  can be estimated by superimposing the rate of retrials to the original departure rate of PROBE packets  $e_m$  by

$$e_m^l = e_m \sum_{n=1}^l (rL_m)^{n-1}. \quad (19)$$

In Eqs. (6)–(8), the  $e_m$  variable is replaced with  $e_m^l$  to take the extra traffic due to retrials into account. In the RLA method,  $L_m$  is first initialized to zero for all the node pairs, so the initial value of  $e_m^l$  equals  $e_m$ . The next value of  $e_m^l$  is calculated by using Eq. (19) and the  $L_m$  value from the fourth step of the RLA algorithm and used in the next iteration. The algorithm continues until  $L_m$  converges. Finally, we calculate the total blocking rate  $L_m^l$  with retrial by subtracting the total success rate from one by

$$L_m^l = 1 - (1 - L_m) \sum_{n=1}^l (rL_m)^{n-1}. \quad (20)$$

### D. Derivation of Transfer Delays

Let  $T_m^l$  be the average transfer time of a successfully reserved flow on a node pair  $m$  with a limit of  $l$  reservation attempts.  $T_m^l$  is the sum of the average time spent for successful reservation  $R_m^l$  and the mean occupation time of data transfer  $\varphi$ . In case there is no retrial of failed reservation attempts,  $R_m^1$  equals to the round-trip time (RTT) of the node pair, so we can say

$$T_m^1 = R_m^1 + \varphi = dD + \varphi. \quad (21)$$

If retrial is allowed, a connection attempt may fail multiple times until it succeeds. Let  $N_m$  be the expected number of hops that the reservation packet passes by in case a reservation attempt fails:

$$N_m = \frac{q_{m,W}^1 + \sum_{n=2}^{d-1} n(q_{m,W}^n - q_{m,W}^{n-1}) + d(L_m - q_{m,W}^{d-1})}{L_m}, \quad (22)$$

where the blocking probability at each hop is weighted by the number of hops that the reservation packets should traverse in case blocking occurs on that hop. The last term in the numerator of Eq. (22) implies that the reservation packet traverses all the links along the path in case a forward blocking occurs in the final hop or a backward blocking occurs.

We assume that network conditions at reservation attempts are uncorrelated. However, link utilizations in a real network may fluctuate in time and cause temporary congestions on a link, which may affect multiple retrials and increase the blocking rate if the retrial rate is higher than the fluctuation speed. It may help to wait by a back-off time  $B$  before trying again to reduce the blocking probability. Our model does not take back-off time into account when calculating the blocking rate, but we can include it in the calculation of transfer delays assuming that back-off delay does not change the blocking probabilities. If the  $n$ th trial of a reservation request succeeds, the average time spent for reservation due to packet exchange and back-off is

$$A_m^n = dD + (n - 1)(N_m D + B), \quad (23)$$

and the probability of succeeding a reservation in this trial is

$$P_m^n = (1 - L_m)(rL_m)^{n-1}. \quad (24)$$

When there is a limit of  $l$  reservation attempts each with a probability of  $r$ , the mean time spent for a successful reservation is

$$\begin{aligned} R_m^l &= \frac{\sum_{n=1}^l A_m^n P_m^n}{1 - L_m} \\ &= \frac{\sum_{n=1}^l (dD + (n - 1)(N_m D + B))(rL_m)^{n-1}}{\sum_{n=1}^l (rL_m)^{n-1}}, \end{aligned} \quad (25)$$

where  $A_m^n$  is weighted by  $P_m^n$  at all possible trial counts by combining Eqs. (20), (23), and (24). Finally,  $T_m^l$  the average transfer time of a successfully reserved flow is calculated by adding  $\varphi$  to  $R_m^l$ .

### III. NUMERICAL RESULTS

We evaluated the performance of the proposed analytical method on the EON topology with the 19 nodes and 39 bidirectional links shown in Fig. 2. Each link carried 16 wavelengths in both directions. The hop delay depends on the propagation delay due to link distance and processing delays influenced by the switching technology used and hardware speed [22], so we tried both a 1 ms hop delay, which means very fast processing and a short link distance, and a 10 ms hop delay, which seemed a more appropriate value. For the sake of simplicity, we used the same hop delay for reservation packets and data flow on each hop in this paper, but they can be assigned different values if necessary. The flow holding time had a mean value of 0.1 s. The Erlang loss model is insensitive to flow holding time distribution, but we applied an exponential distribution for easier simulation. The retrial probability  $r$  was set to 1. We applied the traffic matrix in Table I, which we created by assuming that each node is a gateway for all of the European international traffic for their country and the traffic for two countries is proportional to the product of the two countries' populations. We used the population data of the year 2011 published by The World Bank [23] to create the traffic matrix. Flows between each node pair arrived according to a Poisson process. Some hybrid architectures in the literature try path reservation for only long flows (elephants) larger than a crossover file size, while short flows (mice) are directly sent to the packet network. They try to estimate the optimum crossover file size. We left the traffic splitting as a future work, so all flows try path reservation independent of their size in the following results. The total number of reservation requests in the simulation was  $4 \times 10^9$ , where the first  $4 \times 10^8$  requests were discarded from the results.

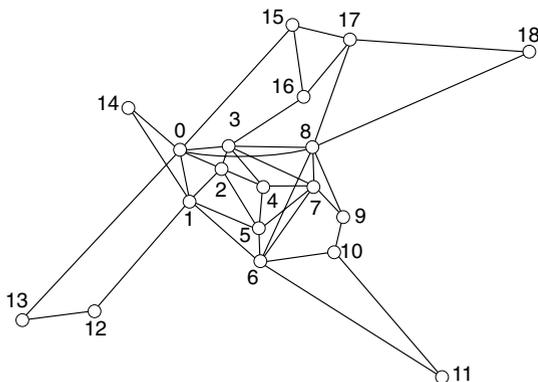


Fig. 2. EON topology.

#### A. Simulation Methodology

We did the simulations by using a discrete event simulator that we wrote in C++. Initially, the simulator calculates the routes between source–destination (s–d) pairs. When there is a connection request, the nodes start exchanging control packets for reservation. Each packet is an object containing the packet headers and the information required by DIR algorithm. The intermediate nodes update the information in the packets before forwarding them. Finally, the destination node reads the packet information and destroys the packet.

As this is a discrete event simulation, the simulator should carry out many packet switching, reading, destroying, and scheduling operations for each reservation request. Moreover, as some node pairs have low blocking rate or traffic rate, the simulator should handle many reservation requests ( $4 \times 10^9$  for each simulation in this paper) in the network in order to get reliable results, so the simulations take a long time.

#### B. Blocking Rate

Many works in the literature report only the total blocking rate in the network as a result, but the total rate may be misleading because analytical and simulation results of individual s–d pairs may have high deviation while giving a close result when the networkwide average is calculated. Therefore, first we report the results of all s–d pairs for greater insight. Figure 3 plots the analytical and simulation results sorted in descending order according to the simulation result of the blocking rate for each s–d pair in the network. Analytical results were added to the graph by matching the same s–d pair in the simulation results. The  $x$  axis shows the s–d pair index. There were 342 s–d pairs in the network. The  $y$  axis shows the average blocking rate, where 1 means 100% blocking. The analytical result by the model proposed in [10] is included in the figure for comparison. The reservation protocol in [10] had a small difference causing 0.5D deviation in the reservation time calculations. We converted it to our reservation time calculation method for a fair comparison.

Figure 3 shows the results when the total reservation request arrival rate in the network was 30 flows/s and the hop delay was 10 ms. Blocked connection attempts were dropped without retrial. Figure 3(a) shows the blocking rate in a linear scale. Small blocking probabilities were difficult to see in the linear scale, so we plotted the same graph in log scale in Fig. 3(b). Some blocking probabilities at the right-hand side of the figure were too low to plot, so we limited the  $x$ -scale to plot only the first 280 s–d pairs. Simulation and analytical results are denoted by (S) and (A), respectively. The total blocking rate and the forward blocking rate are denoted by total and forward in the figure, respectively. The figure reveals that the result of our analytical calculation matched the simulation result very well. However, the analysis by [10] had a high error when compared with the simulation results. As the forward

TABLE I  
EON TRAFFIC MATRIX

	0	1	2	3	4	5	6	7	8	9	
London	0	0	0.014804	0.00249	0.003777	0.000117	0.001789	0.013748	0.002386	0.018489	0.001905
Paris	1	0.014804	0	0.002601	0.003946	0.000122	0.001869	0.014361	0.002492	0.019314	0.00199
Brussels	2	0.00249	0.002601	0	0.000664	2.06E-05	0.000314	0.002416	0.000419	0.003249	0.000335
Amsterdam	3	0.003777	0.003946	0.000664	0	3.12E-05	0.000477	0.003664	0.000636	0.004928	0.000508
Luxembourg	4	0.000117	0.000122	2.06E-05	3.12E-05	0	1.48E-05	0.000113	1.97E-05	0.000153	1.57E-05
Zurich	5	0.001789	0.001869	0.000314	0.000477	1.48E-05	0	0.001735	0.000301	0.002334	0.00024
Milan	6	0.013748	0.014361	0.002416	0.003664	0.000113	0.001735	0	0.002315	0.017936	0.001848
Prague	7	0.002386	0.002492	0.000419	0.000636	1.97E-05	0.000301	0.002315	0	0.003113	0.000321
Berlin	8	0.018489	0.019314	0.003249	0.004928	0.000153	0.002334	0.017936	0.003113	0	0.002485
Vienna	9	0.001905	0.00199	0.000335	0.000508	1.57E-05	0.00024	0.001848	0.000321	0.002485	0
Zagreb	10	0.000997	0.001041	0.000175	0.000266	8.23E-06	0.000126	0.000967	0.000168	0.001301	0.000134
Athens	11	0.002557	0.002671	0.000449	0.000682	2.11E-05	0.000323	0.002481	0.000431	0.003336	0.000344
Madrid	12	0.01046	0.010926	0.001838	0.002788	8.63E-05	0.00132	0.010147	0.001761	0.013646	0.001406
Lisbon	13	0.002406	0.002514	0.000423	0.000641	1.99E-05	0.000304	0.002335	0.000405	0.00314	0.000323
Dublin	14	0.001015	0.00106	0.000178	0.000271	8.38E-06	0.000128	0.000985	0.000171	0.001324	0.000136
Oslo	15	0.00112	0.00117	0.000197	0.000299	9.25E-06	0.000141	0.001087	0.000189	0.001462	0.000151
Copenhagen	16	0.001261	0.001317	0.000222	0.000336	1.04E-05	0.000159	0.001223	0.000212	0.001645	0.000169
Stockholm	17	0.002139	0.002234	0.000376	0.00057	1.77E-05	0.00027	0.002075	0.00036	0.00279	0.000287
Moscow	18	0.032108	0.033541	0.005642	0.008558	0.000265	0.004053	0.031149	0.005406	0.041891	0.004315
	10	11	12	13	14	15	16	17	18		
London	0	0.000997	0.002557	0.01046	0.002406	0.001015	0.00112	0.001261	0.002139	0.032108	
Paris	1	0.001041	0.002671	0.010926	0.002514	0.00106	0.00117	0.001317	0.002234	0.033541	
Brussels	2	0.000175	0.000449	0.001838	0.000423	0.000178	0.000197	0.000222	0.000376	0.005642	
Amsterdam	3	0.000266	0.000682	0.002788	0.000641	0.000271	0.000299	0.000336	0.00057	0.008558	
Luxembourg	4	8.23E-06	2.11E-05	8.63E-05	1.99E-05	8.38E-06	9.25E-06	1.04E-05	1.77E-05	0.000265	
Zurich	5	0.000126	0.000323	0.00132	0.000304	0.000128	0.000141	0.000159	0.00027	0.004053	
Milan	6	0.000967	0.002481	0.010147	0.002335	0.000985	0.001087	0.001223	0.002075	0.031149	
Prague	7	0.000168	0.000431	0.001761	0.000405	0.000171	0.000189	0.000212	0.00036	0.005406	
Berlin	8	0.001301	0.003336	0.013646	0.00314	0.001324	0.001462	0.001645	0.00279	0.041891	
Vienna	9	0.000134	0.000344	0.001406	0.000323	0.000136	0.000151	0.000169	0.000287	0.004315	
Zagreb	10	0	0.00018	0.000736	0.000169	7.14E-05	7.88E-05	8.87E-05	0.00015	0.002259	
Athens	11	0.00018	0	0.001888	0.000434	0.000183	0.000202	0.000228	0.000386	0.005794	
Madrid	12	0.000736	0.001888	0	0.001776	0.000749	0.000827	0.000931	0.001578	0.023699	
Lisbon	13	0.000169	0.000434	0.001776	0	0.000172	0.00019	0.000214	0.000363	0.005452	
Dublin	14	7.14E-05	0.000183	0.000749	0.000172	0	8.02E-05	9.03E-05	0.000153	0.0023	
Oslo	15	7.88E-05	0.000202	0.000827	0.00019	8.02E-05	0	9.97E-05	0.000169	0.002538	
Copenhagen	16	8.87E-05	0.000228	0.000931	0.000214	9.03E-05	9.97E-05	0	0.00019	0.002857	
Stockholm	17	0.00015	0.000386	0.001578	0.000363	0.000153	0.000169	0.00019	0	0.004845	
Moscow	18	0.002259	0.005794	0.023699	0.005452	0.0023	0.002538	0.002857	0.004845	0	

blocking was very low due to low utilization, it is not possible to see it in the figure and almost all blocking was due to backward blocking in the simulation. Likewise, many core networks on the Internet are run at low utilization levels, so most blocking occurs in the backward direction unless the RTTs are very low. Therefore, the accuracy of the backward blocking estimation is extremely important. Our proposed model for backward blocking analysis calculated the backward blocking with high accuracy, so the total blocking rates by simulations and our analysis in Fig. 3 were close to each other.

The analytical calculation of our result in Fig. 3 took around 1.3 s on a single core of an Intel Core i7-3960X CPU by using a not so optimized and single-threaded program written in C++. It seems possible to compute the same analytical result in less than 0.1 s by using a well-optimized, multithreaded program on the same CPU. In comparison, the simulation in the same figure took more than 4 h. The results of analytical method and simulations

were close to each other in Fig. 3, while the analytical method was several orders of magnitude faster than the simulation.

Most of the analytical methods in the literature have problems when estimating the blocking rate on highly loaded links. Fortunately, many core networks on the Internet are operated at low loads, but we also present a high load example as a reference. Moreover, recently more WDM networks are modestly or heavily utilized, so an analytical model that can calculate blocking rates in congested networks can be beneficial. We increased the network load by increasing the total reservation arrival rate to 500 flows/s, which caused a maximum of 67% blocking rate between the s-d pair (18-12). Again, the result of our analytical calculation matched the simulation result very well as shown in Fig. 4. A comparison of the two analytical methods revealed that the result of our analysis was much closer to the simulation results than was the model in [10]. Forward blocking rates became visible in the figure, but

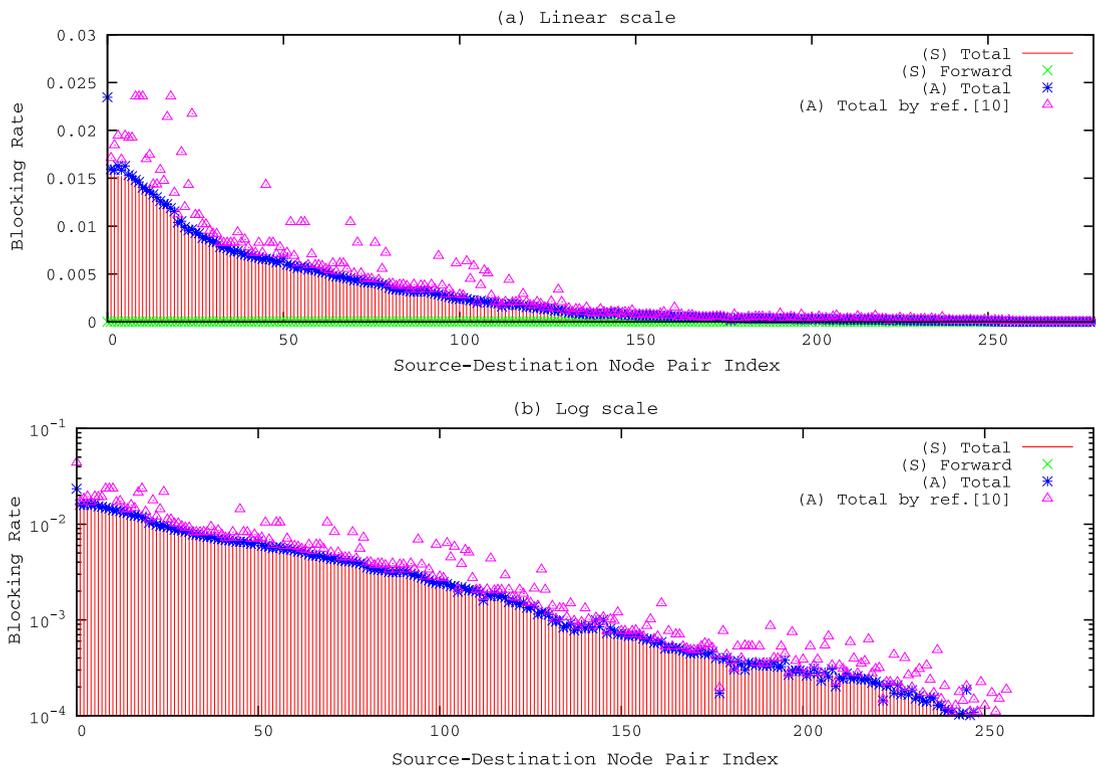


Fig. 3. Blocking rate when the request arrival rate was 30 flows/s and the hop delay was 10 ms.

their ratio was still low compared to the total blocking. Figure 4 shows that for many node pairs with similar total blocking rate, the ratio of backward and forward blocking

rates are much different from each other. The reason is that the forward blocking is caused by insufficient network capacity, while the backward blocking is caused by

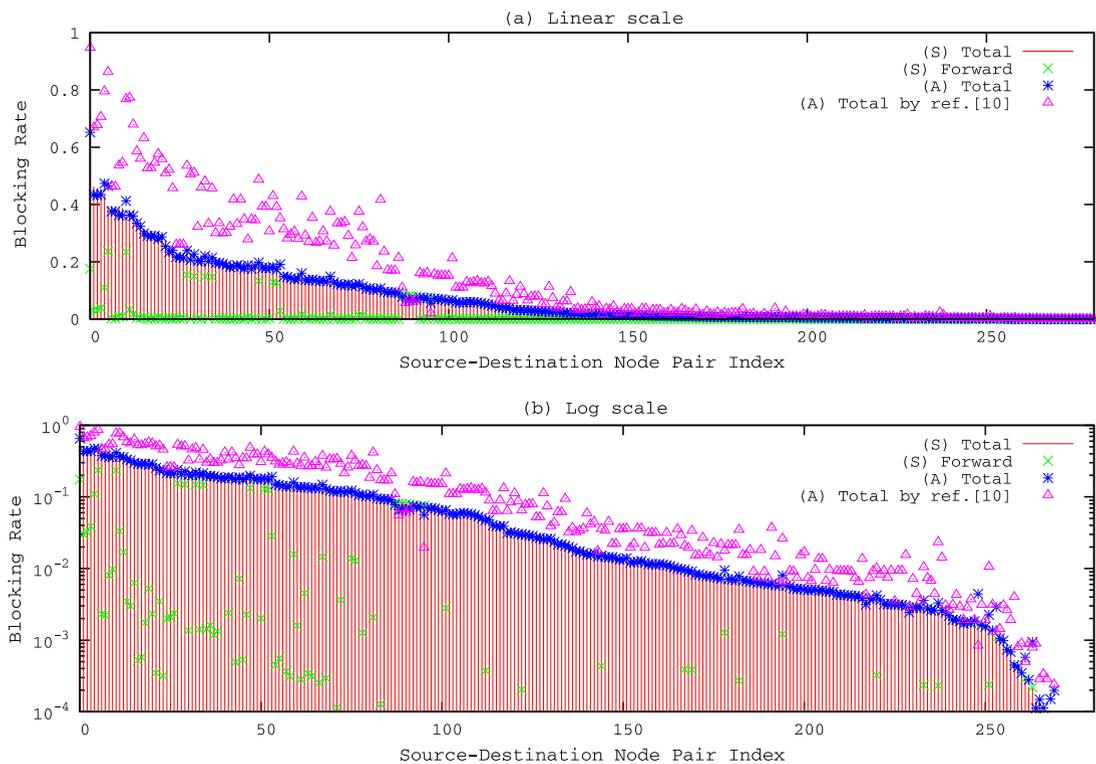


Fig. 4. Blocking rate when the request arrival rate was 500 flows/s and the hop delay was 10 ms.

outdated information. Even though the total blocking rate of some s–d pairs are similar, their RTTs and the distribution of utilization rates along their path are different, which causes different forward and backward blocking rates.

A highly loaded network with high forward blocking is an extremely difficult scenario for an analytical calculation. We further increased the percentage of forward blocking by decreasing the hop delay to 1 ms while keeping the arrival rate at 500 flows/s to analyze such an extreme scenario. As the hop delay was lower, the reservation packets carried more recent information about wavelength occupancy, so the backward blocking rate decreased considerably. The simulation results in Fig. 5 show that our analytical calculation still matched the simulation result with a low error rate. Figure 5 reveals that even in such an extreme scenario, a high proportion of the blocking in more than half of the s–d pairs was due to the backward blocking, so the accuracy of the backward blocking analysis has a great impact on the accuracy of the overall blocking calculation.

As the next step, we evaluated the analytical model with retri- al of failed reservation attempts. Figure 6 shows the blocking rate when there was no retri- al (R:0) and when there was a maximum of two retri- als (R:2), which means that the reservation algorithm attempted to do a reservation a maximum of three times. The total reservation request arrival rate was 500 flows/s and the hop delay was 10 ms. Again this example uses a highly loaded network, which is a difficult scenario for analytical calculation. However, high load is necessary for comparing the results

of retri- al, as the blocking rate becomes extremely low and hard to compare when the network load is low. The back-off time between trials was set to zero (B:0) and 1000 ms (B:1000) in the simulations. Figure 6 shows that retri- al can greatly decrease the overall blocking rate. We see that there was around a 1000-times decrease in the blocking rate at the right-hand side of the figure. When we compare the simulation and analytical results with two retri- als, we see that the simulation with 1000 ms back-off time gave slightly lower blocking probability than the simulation without back off. The reason is that temporary congestion due to fluctuations on link utilization may cause the subsequent retri- als to be blocked if the retri- al rate is higher than the fluctuation speed. The analytical model assumes there is no correlation between network states at retri- al times, so the analytical result is closer to the simulation with 1000 ms back off in general, which decreases the correlation. Because of the recursive nature of the retri- als as they are superimposed on the traffic matrix, the difference between the simulation and analytical results is magnified with each retri- al, which can be seen at the blocking rate results of some node pairs between 90 and 100 in Fig. 6.

Next, we show the load distribution in the network for the scenarios with 30 and 500 flows/s traffic rate, each with two retri- als and without retri- al, when the hop delay was 10 ms and the back-off delay was zero. Figure 7 plots the average wavelength utilization rate of links in simulation, where the  $x$  axis shows the links represented as an ordered pair of nodes, in the form from node  $i$  to node  $j$ , denoted by  $i - j$ . The links in the  $x$  axis are sorted according to their link utilization level in the scenario with 30 flows/s

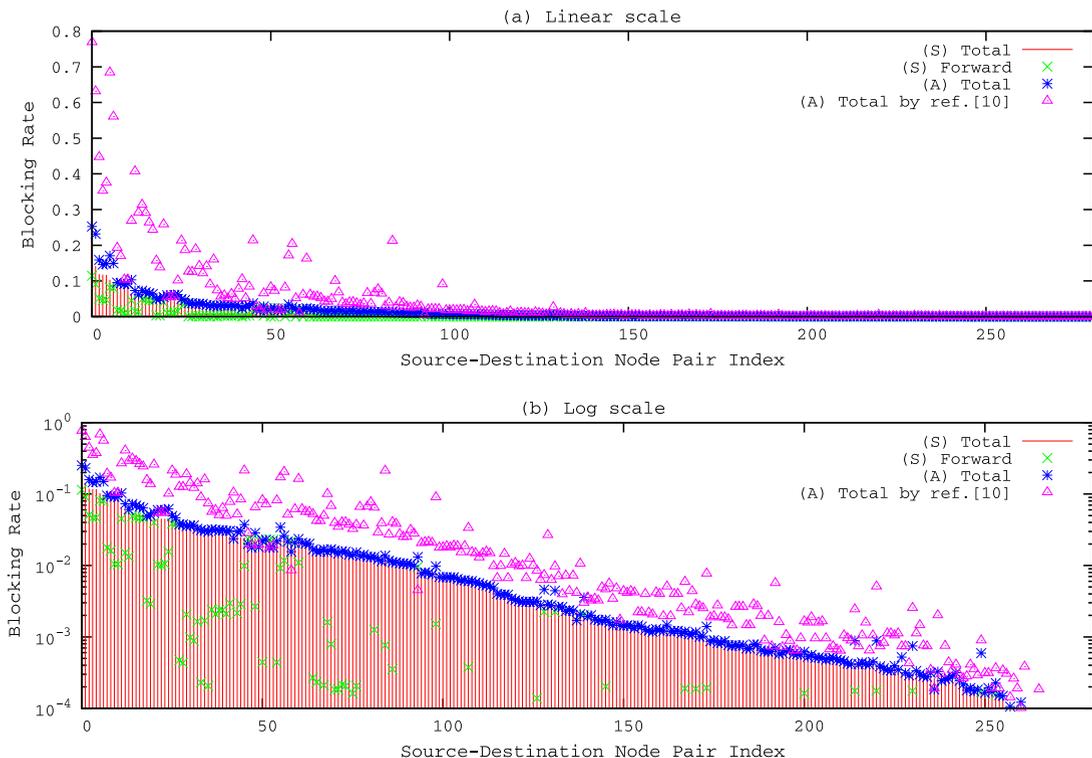


Fig. 5. Blocking rate when the request arrival rate was 500 flows/s and the hop delay was 1 ms.

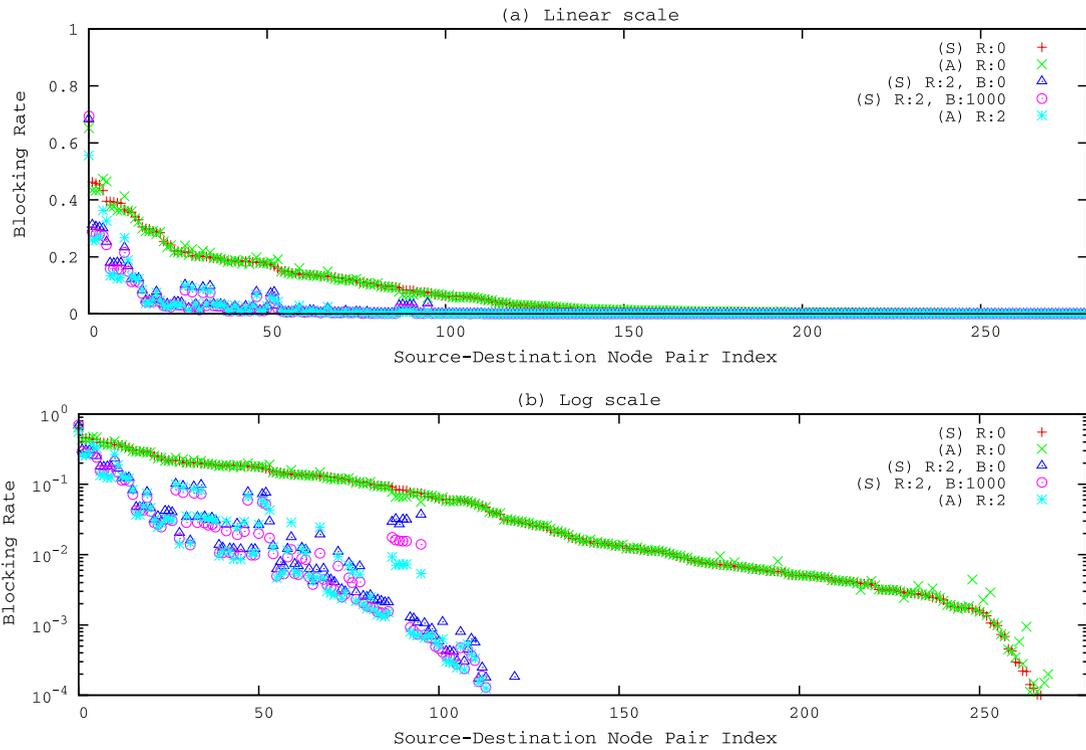


Fig. 6. Blocking rate with retrial when the total reservation request arrival rate was 500 flows/s and the hop delay was 10 ms.

traffic and no retrial. Figure 7 reveals that the link from node 8 to 18 had the highest utilization in all scenarios. When the traffic rate was 30 flows/s, the average link load in the network was 0.635% for the case without retrials and 0.639% for the case with a maximum of two retrials. The extra traffic due to retrials was low, so their plots almost overlap in Fig. 7. When the traffic rate was 500 flows/s, the average link load in the network increased to 8.88% in the case without retrials and 10.02% in the case with a maximum of two retrials. The retrial caused up to 20% increase in the wavelength utilization of links, where the most congested link (8–18) had a utilization of 76.1% without retrial and 86.1% with retrial.

It is important to assess the accuracy of an analytical model for varying loads and show that the analysis is

applicable to other topologies. Therefore, we added results on NSFNET topology with 14 nodes and 21 bidirectional links, a ring topology (Ring-5) with five nodes and five bidirectional links and a ring topology (Ring-11) with 11 nodes and 11 bidirectional links. We calculated the networkwide blocking rates for a wide range of traffic arrival rates from 10 to 3500 flows/s when the hop delay was 10 ms. We applied the traffic demand matrix in [24] to the NSFNET topology. The ring topologies used a uniform traffic matrix. The other parameters were the same as the EON topology.

Figure 8 shows the total blocking rate versus traffic arrival rate for comparing simulation and analytical results when there was no retrial and when there was a maximum of two retrials with 0 and 1000 ms back off. Moreover, the results of the forward blocking rate by simulation are

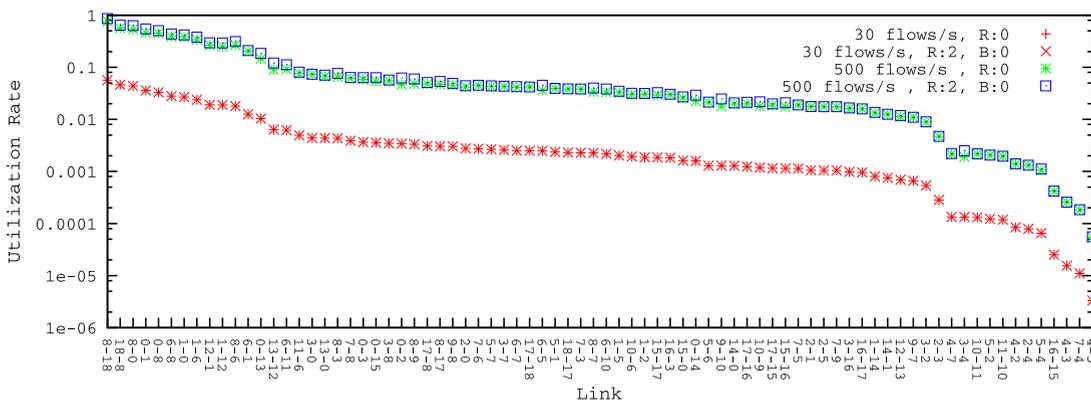


Fig. 7. Wavelength utilization rate when the hop delay was 10 ms.

plotted in the same figure to show the ratio of backward and forwarding blocking. The total blocking rate was calculated by weighting the overall blocking rate of each node pair by its traffic arrival rate. The total blocking rate with retrials is the resulting blocking rate after trials, but the forward blocking rate is the result of a single trial. As seen in Fig. 8, the simulation and analytical results of total blocking without retrial matched very well in all topologies. When the maximum retrial rate was two, the simulations without back-off delay gave a little higher total blocking rate than the simulations with 1000 ms back-off delay because of the higher correlation between network states at retrial times when there is no back-off delay. Among the tested topologies, Ring-5 topology had the most significant change in blocking rate with back off, where the back off decreased the total blocking rate up to 4 times. The analytical method assumes that there is no correlation between network states at retrial times, so the analytical result was close to the simulation results with back off. In Ring-11 topology, the analytical result with retrial was a bit higher than the simulation results. The reason is that the Ring-11 topology had a maximum hop length of five, which was the highest among the tested topologies. As the analytical method calculates the blocking rate in a recursive manner along the path, the estimation error accumulates and increases with the number of hops. The retrials further increase the error rate exponentially.

Therefore, the Ring-11 topology had a bit higher deviation between analytical and simulation results of the retrial scenario compared to the other topologies.

When we compare the total and forward blocking rates without retrial in Fig. 8, we see that most of the blocking was due to backward blocking when the traffic arrival rate was low. The forward blocking got the majority only when the traffic rate and, thus, the total blocking rate was high. Again, this shows the importance of the accuracy of a backward blocking analysis when calculating the total blocking in low loaded networks. Our proposed backward blocking analysis was highly accurate as seen in the low blocking regions on the left-hand side of all subfigures in Fig. 8. When retrial was allowed, the forward blocking rate in a single trial considerably increased, because the retrial traffic, which was superimposed on the call setup rate, further increased the load and wavelength utilization in the network, making it more difficult to find a wavelength that satisfies the wavelength-continuity constraint.

### C. Reservation Delay

In a hybrid network architecture, one of the aims is to minimize the overall flow completion time, so the reservation delay is an important metric for a path switching layer.

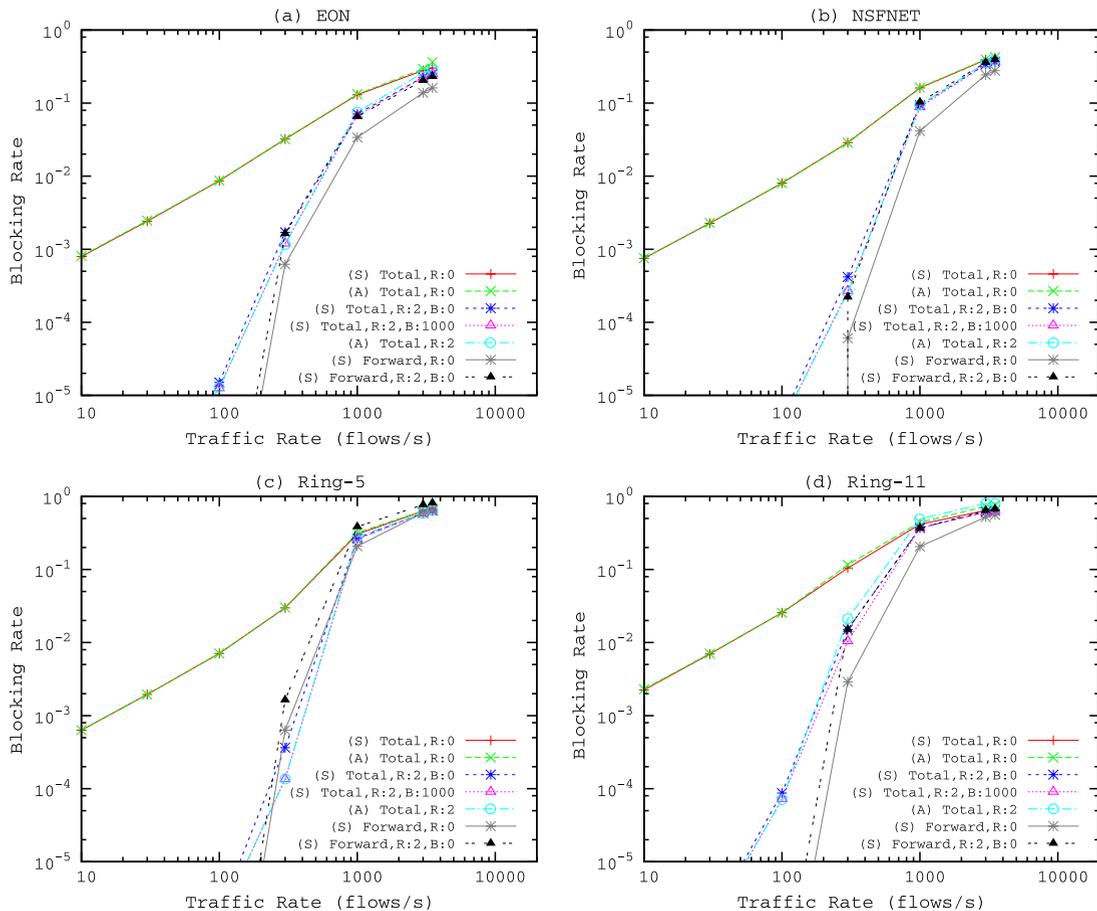


Fig. 8. Blocking rate versus load in (a) EON, (b) NSFNET, (c) five-node ring, and (d) 11-node ring topologies.

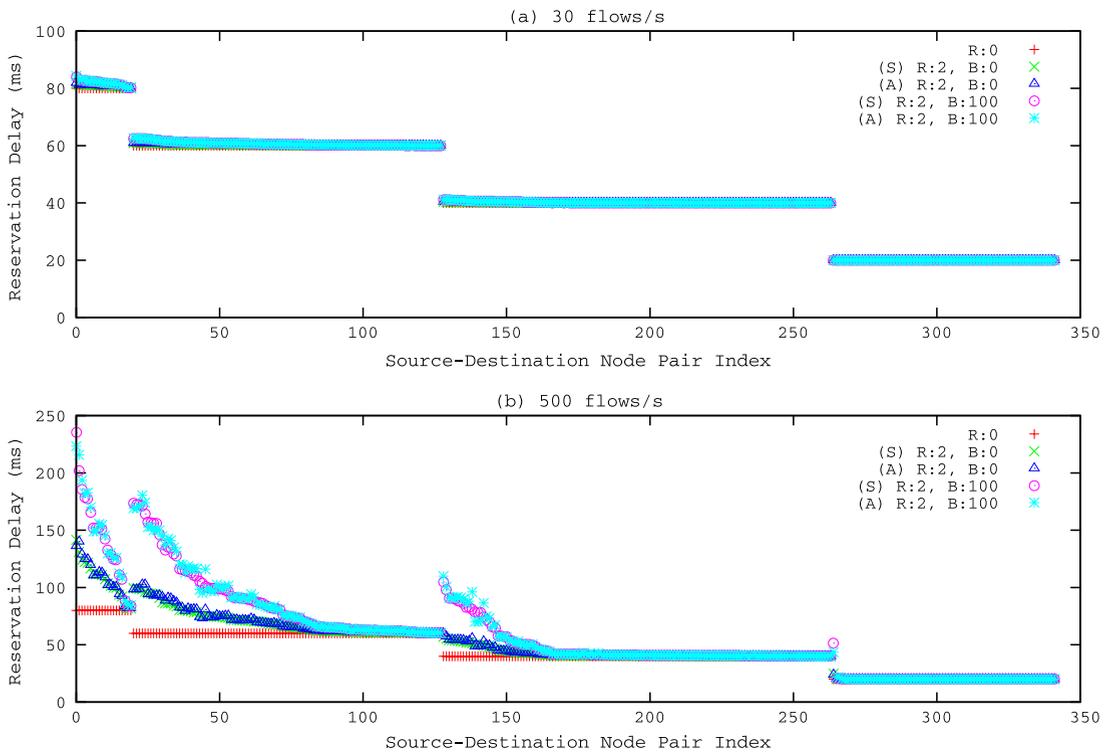


Fig. 9. Reservation delay of flows, which successfully reserve a path, when the total reservation request arrival rate is (a) 30 flows/s and (b) 500 flows/s.

Figure 9 shows the reservation delay of flows, which successfully reserved a path, when the total reservation request arrival rate was 30 flows/s and 500 flows/s. The hop delay was 10 ms. When there was retrial, we tried 0 and 100 ms back-off time, which was selected to be equal to the average flow size. When there was no retrial, the reservation delay was simply the RTT of the path. Figure 9(a) shows that retrial and 100 ms back off did not increase the reservation delay much when the blocking rate is low, but they substantially increased the time spent for reservation when the blocking rate was high as seen in Fig. 9(b). On the most left-hand side of Fig. 9(b) for 500 flows/s traffic, the reservation delay was almost doubled with retrial in case of zero back off and tripled in case of 100 ms back off. Even though the retrial and back off increased the efficiency of the path network and, thus, decreased the need for a packet network, it also increased the flow completion time in the path network. Therefore, the maximum retrial count and back-off time should be selected carefully when the blocking rate is high. When we compare the simulation and analytical results, we see that the difference was low at both 30 and 500 flows/s traffic rate.

#### IV. CONCLUSIONS

In this paper, we proposed an analytical method with an improved backward blocking analysis based on RLA for calculating blocking probabilities and reservation delays in path switching optical WDM networks with DIR with retrial. Such an analytical method can be useful in the fast

calculation of these two metrics for traffic engineering and optimization of the traffic-splitting parameters for hybrid optical architectures combining path and packet switching. We compared the analytical and simulation results on a mesh EON network and showed that their results were close to each other, while our analytical method was several orders of magnitude faster than the simulation, which may allow much faster performance analysis, design, and optimization of hybrid networks. We showed that even in congested networks a high proportion of the blocking is due to the backward blocking, so the accuracy of our proposed backward blocking analysis had a great impact on the accuracy of the overall blocking calculation. The impact of assumptions in the analytical model was found to be small unless the blocking rate was too high.

As future work, we will work on a hybrid path-packet integrated architecture that makes use of our analytical method for traffic splitting.

#### ACKNOWLEDGMENTS

This work was supported by the National Institute of Information and Communications Technology (NICT).

#### REFERENCES

- [1] H. Harai, "Optical packet and circuit integrated network system and testbed," invited talk presented at the ECOC 2010 Co-located Workshop, Torino, Italy, Sept. 2010 [Online]. Available: [http://www.ecoc2010.org/contents/attached/c20/WS\\_2\\_Harai.pdf](http://www.ecoc2010.org/contents/attached/c20/WS_2_Harai.pdf).

- [2] S. Arakawa, N. Tsutsui, and M. Murata, "A biologically-inspired wavelength resource allocation for optical path/packet integrated networks," in *Proc. 15th Conf. Optical Network Design and Modeling*, Feb. 2011.
- [3] E. Van Breusegern, J. Cheyens, D. De Winter, D. Colle, M. Pickavet, F. De Turck, and P. Demeester, "Overspill routing in optical networks: A true hybrid optical network design," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 4, pp. 13–25, 2006.
- [4] S. Bjornstad, D. Hjelm, and N. Stol, "A packet-switched hybrid optical network with service guarantees," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 97–107, Aug. 2006.
- [5] X. Zheng, M. Veeraraghavan, N. S. V. Rao, Q. Wu, and M. Zhu, "CHEETAH: Circuit-switched high-speed end-to-end transport architecture testbed," *IEEE Commun. Mag.*, vol. 43, no. 8, pp. S11–S17, Aug. 2005.
- [6] X. Yuan, R. Melhem, R. Gupta, Y. Mei, and C. Qiao, "Distributed control protocols for wavelength reservation and their performance evaluation," *Photon. Netw. Commun.*, vol. 1, no. 3, pp. 207–218, 1999.
- [7] L. Berger, "Generalized multi-protocol label switching (GMPLS) signaling resource reservation protocol-traffic engineering (RSVP-TE) extensions," IETF RFC 3473, Jan. 2003.
- [8] L. Berger, ed., "Generalized multi-protocol label switching (GMPLS) signaling functional description," IETF RFC 3471, Jan. 2003.
- [9] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: A novel approach to high bandwidth optical WANs," *IEEE Trans. Commun.*, vol. 40, no. 7, pp. 1171–1182, July 1992.
- [10] S. Arakawa, K. Miyamoto, M. Murata, and H. Miyahara, "Performance analyses of wavelength reservation methods for high-speed data transfer in photonic networks," in *Proc. ITC-CSCC'99*, San Antonio, TX, July 1999, pp. 828–831.
- [11] R. B. Cooper and S. Katz, "Analysis of alternative routing networks with account taken of nonrandomness of overflow traffic," Bell Telephone Lab. Tech. Rep., 1964.
- [12] S. P. Chung, A. Kashper, and K. W. Ross, "Computing approximate blocking probability for large loss networks with state-dependent routing," *IEEE/ACM Trans. Netw.*, vol. 1, no. 1, pp. 105–115, Feb. 1993.
- [13] A. Birman, "Computing approximate blocking probabilities for a class of all-optical networks," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 5, pp. 852–857, June 1996.
- [14] M. Kovacevic and A. S. Acampora, "Benefits of wavelength translation in all-optical clear-channel networks," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 5, pp. 868–880, June 1996.
- [15] A. Sridharan and K. N. Sivarajan, "Blocking in all-optical networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 2, pp. 384–397, Apr. 2004.
- [16] Y. Zhu, G. N. Rouskas, and H. G. Perros, "A path decomposition approach for computing blocking probabilities in wavelength-routing networks," *IEEE/ACM Trans. Netw.*, vol. 8, no. 6, pp. 747–762, Dec. 2000.
- [17] J. P. Jue and G. Xiao, "Analysis of blocking probability for connection management schemes in optical networks," in *Proc. IEEE GLOBECOM*, San Antonio, TX, Nov. 2001, pp. 1546–1550.
- [18] K. Lu, G. Xiao, and I. Chlamtac, "Analysis of blocking probability for distributed lightpath establishment in WDM optical networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 1, pp. 187–197, Feb. 2005.
- [19] O. Alparslan, S. Arakawa, and M. Murata, "Computing path blocking probabilities for traffic splitting in optical hybrid switching networks," in *Proc. IEEE Int. Conf. Communications*, Ottawa, Canada, June 2012.
- [20] F. Xue, S. Yoo, H. Yokoyama, and Y. Horiuchi, "Performance analysis of wavelength-routed optical networks with connection request retrials," in *Proc. IEEE Int. Conf. Communications*, May 2005, pp. 1813–1818.
- [21] J. Cao, W. S. Cleveland, D. Lin, and D. X. Sun, "On the non-stationarity of Internet traffic," in *Proc. SIGMETRICS/Performance'01*, 2001, pp. 102–112.
- [22] G. Papadimitriou, C. Papazoglou, and A. Pomportsis, "Optical switching: Switch fabrics, techniques, and architectures," *J. Lightwave Technol.*, vol. 21, no. 2, pp. 384–405, Feb. 2003.
- [23] Population. [Online]. Available: <http://data.worldbank.org/indicator/SP.POP.TOTL>.
- [24] R. Ramaswami and K. N. Sivarajan, "Design of logical topologies for wavelength-routed optical networks," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 5, pp. 840–851, June 1996.